

Linguistic Issues in Language Technology – **LiLT**

**Detection and Resolution of Verb
Phrase Ellipsis**

Marjorie McShane

Petr Babkin

Published by CSLI Publications

Detection and Resolution of Verb Phrase Ellipsis

MARJORIE MCSHANE, *Rensselaer Polytechnic Institute*,
PETR BABKIN, *Rensselaer Polytechnic Institute*

Abstract

Verb phrase (VP) ellipsis is the omission of a verb phrase whose meaning can be reconstructed from the linguistic or real-world context. It is licensed in English by auxiliary verbs, often modal auxiliaries: *She can go to Hawaii but he can't [e]*. This paper describes a system called ViPER (**VP Ellipsis Resolver**) that detects and resolves VP ellipsis, relying on linguistic principles such as syntactic parallelism, modality correlations, and the delineation of core vs. peripheral sentence constituents. The key insight guiding the work is that not all cases of ellipsis are equally difficult: some can be detected and resolved with high confidence even before we are able to build systems with human-level semantic and pragmatic understanding of text.

1 Introduction

Verb phrase (VP) ellipsis is the omission of a VP whose meaning can be reconstructed from the linguistic or real-world context. For example, the clause *he could* in (1) cannot be fully interpreted without reconstructing the elided VP as *teach illiterate women to read and write*.¹

- (1) Aid workers in war-ravaged Kabul were stunned when a toddler from a poor family offered to *teach illiterate women to read and write* – and then promptly proved he could [e].

Ellipsis should be resolved by language processing systems because unresolved ellipses mask information that is available to a human reader.

Ellipsis has not been treated much by NLP systems. Bos and Spina 2011 assess why:

“First, from a purely practical perspective, automatically locating ellipsis and their antecedents is a hard task, not subsumed by ordinary natural language processing components. Recent empirical work (Hardt 1997; Nielsen 2005) indeed confirms that VPE identification is difficult. Second, most theoretical work begins at the point at which the ellipsis example and the rough location of its antecedent are already identified, focussing on the resolution task” (pp. 464-465).

Certainly, both of the subtasks of ellipsis processing – detection and resolution – can be very difficult, so it is unrealistic to expect a full-coverage, near-term solution for open text. However, although some instances of VP ellipsis require human-level reasoning, not all are so difficult. In fact, it is possible to automatically detect which cases are simpler and resolve them with high precision, which is exactly what ViPER (**VP Ellipsis Resolver**) does. So, like many NLP systems, ViPER treats only a subset of instances; but unlike many systems, ViPER automatically selects the subset it knows how to treat and processes those instances fully automatically, without the need for manual corpus annotation.

ViPER pursues string-level resolution of VP ellipsis, defined as copying the sponsor string to fill the elliptical gap. This is a midstream result toward full meaning analysis; and, as a midstream result, it has

¹In examples, elided categories are indicated by [e] and their sponsors – i.e., the material used to recover their meaning – are italicized. We use the term “sponsor” rather than “antecedent” because “antecedent” has two unnecessary implications: that the sponsor is a linguistic entity, and that it precedes the referring expression. All numbered examples, except those labelled “invented”, are from the Gigaword corpus (Graff and Cieri 2003). Although examples included in the evaluation could include mismatched punctuation – e.g., opening quotation marks that are not closed within the excerpt – we normalized such punctuation in order not to distract readers’ attention with irrelevant details.

some suboptimal features. First, sponsors for elided referring expressions – like sponsors for any referring expressions – are most accurately understood as disambiguated semantic entities rather than potentially ambiguous text strings. Second, it can be impossible to select a text string that fully and precisely reflects the reconstructed meaning. For example in (2), a string-level reconstruction would be *appeal the ruling*, which is not directly available in the context.

- (2) The ruling can be appealed, but it was not immediately clear if al-Dossari would [e].

Third, string-based resolution does not make the crucial semantic distinction between type- and instance-coreference between verbs, or strict and sloppy coreference between their internal arguments.² For example, in (3) the elided category actually refers to a different instance of sending a different set of children abroad than is presented in the sponsor clause.

- (3) Better-off parents could *send their children abroad for English education* but poorer families could not [e].

However, despite the global insufficiencies of a string-based approach, it offers three advantages. First, ViPER – or other implementations of the same approach – can be used right away since all necessary preconditions are fulfillable within the current state of the art. Second, string-level resolution can be a useful end result for shallow NLP systems, offering a larger percentage of overt text strings for the systems to manipulate. Finally, string-level resolution offers useful heuristic evidence for systems that pursue full semantic analysis.

Our approach to detecting treatable (meaning “treatable by ViPER”) instances of VP ellipsis orients largely around syntactic simplicity. The hypothesis is that syntactically simple elliptical structures are easier to resolve than syntactically complex ones due to the limited number of verbal elements competing to serve as the sponsor. The following set of examples juxtaposes simple and complex contexts.

- (4) And you try to *get bragging rights* if you can [e].
- (5) The former Massachusetts governor called on United Nations Secretary General Ban Ki-moon to *revoke Ahmadinejad’s invitation to the assembly* and warned Washington should reconsider support for the world body if he did not [e].

²There has been extensive study of sloppy vs. strict coreference in theoretical syntax, dating back at least to Fiengo and May 1994.

- (6) “We’re celebrating the fact that we’re living in a time where, when we want to *be in the kitchen*, we can [e],” says Tamara Cohen, Ma’yan program director.

Example (4) is syntactically simple because the clause containing the sponsor occurs right before the ellipsis clause and there are no additional main verbs competing to be the sponsor. By contrast, both (5) and (6) contain multiple candidate sponsors. (5) is truly difficult, even requiring significant concentration by a person reading the context. ViPER recognizes the complexity of such contexts and does not attempt to treat them. By contrast, (6) would fall into our simple category if the portion preceding the first comma were removed. ViPER carries out a series of syntactic tree trimming processes to render examples like (6) functionally simple and, therefore, treatable.

Once the sponsor clause has been identified, the next challenge is to determine the actual text elements contributing to the ellipsis reconstruction. For example, in (4) the modal element *try to* must be excluded from the sponsor; however, modals are sometimes included in sponsors. To handle modality during VP ellipsis resolution, ViPER uses an extended version of the microtheory of modality formulated within the theory Ontological Semantics (Nirenburg and Raskin 2004). Ontological Semantics distinguishes ten types of modality, at least nine of which can license VP-ellipsis.³ Sample strings conveying each type of modality are: **epistemic**: did, didn’t, will, won’t; **permissive**: may, may not; **epiteuctic**: fail to; **volitive**: want to, not want to; **obligative**: must, must not, has to, doesn’t have to; **potential**: can, can’t, is not able to; **evaluative**: loves to, doesn’t like to; **intentional**: intends to, doesn’t intend to; **effort**: try to, not bother to. Each modal meaning has a scope that represents the proposition to which the modal meaning applies. A proposition can be scoped over by many different types of modal meanings, as shown by (7), and multiple modalities can scope over a proposition, as shown by (8).

- (7) The US has <has not, has to, might have, cannot have, should not, might, wants to, does not want to, seems to have, could not have, is believed to have, failed to, etc.> sign(ed) the treaty.*Invented*
- (8) The US might have to <could want to, etc.> sign the treaty.*Invented*

ViPER carries out modality analysis to determine how the modal verbs in sponsor clauses should be handled during ellipsis resolution. For example, when the modalities in the sponsor clause and ellipsis clause are

³We have not found examples of *belief* modality licensing ellipsis.

semantically paired, as with *try to ... can*, the sponsor-clause modality is excluded from the sponsor.

The remainder of the paper describes how ViPER detects ellipsis (Section 2), selects which instances it can treat (Section 3), and determines the actual sponsor once the sponsor clause has been identified (Section 4). We then report a system evaluation (Section 5), place this contribution within the larger field (Section 6), and describe ongoing and future work (Section 7).

2 Ellipsis Candidate Detector

Detecting VP ellipsis can be tricky. Although the simplest algorithm would be to seek instances of auxiliary verbs that are not directly followed by a verbal string that could serve as their complement, this can fail in at least three circumstances: when the auxiliary is separated from its complement by a parenthetical expression (9), when a string that looks like an auxiliary is actually functioning as a main verb (10), and when a string that looks like an auxiliary verb is actually a nominal (11).

- (9) “I’ve always tried to, when we get intelligence, check with other several sources, ...”
- (10) He has more stamps than she *has*.*Invented*
- (11) They showed their *might*.*Invented*

To handle these challenges, ViPER detects elliptical contexts using a three-stage process.

1. It searches the corpus for auxiliary verbs directly followed by a punctuation mark or conjunction, since these indicate discourse breaks that could indicate that a VP has been elided. For reasons that will become clear shortly, it divides hits into those with a hard discourse break (a period, colon or semi-colon), a soft discourse break (a comma, dash or open-parenthesis), and a conjunction. Naturally, this method will not offer full recall since VP ellipsis need not be followed by a punctuation mark or conjunction; however, it covers many cases and was more than sufficient to support the testing of our linguistic hypotheses about ellipsis resolution.⁴

⁴To improve recall, one could, e.g., parse the entire corpus from the outset and use missing dependencies to detect ellipsis. For example, in the parse of the non-elliptical sentence “John didn’t come but Mary did come” there is an *aux* dependency between the main verb of the second clause, *come*, and the auxiliary *did*. By contrast, in the elliptical counterpart “John didn’t come but Mary did [e]” the *aux* dependency is

2. It parses the candidate elliptical contexts using the Stanford CoreNLP toolkit (version 3.4.1; Manning et al. 2014; referred to hereafter as CoreNLP), which identifies certain kinds of false positives, such as nominals (11) and some parentheticals (9).
3. Since CoreNLP’s detection of parentheticals is incomplete, we use additional parenthetical detection functions to further prune out false positives (see Appendix A for details). Our hypothesis was that if the system could detect parentheticals with high confidence, then it could treat all nonparenthetical cases as elliptical. More specifically, it treats “modal + soft discourse break + nonparenthetical” as if the soft discourse break were a hard discourse break, removing the sentence elements following the soft discourse break and applying ViPER’s ellipsis resolution methods to what remains.

All candidate examples that are not pruned out by these processes are considered elliptical, and ViPER must next determine which of them it will attempt to resolve.

3 Three Resolution Strategies

ViPER uses three strategies to detect which instances of VP ellipsis it can resolve. These strategies are described in turn below.

3.1 Elliptical Phrasal Detector

The first strategy involves configurations that we broadly characterize as phrasals. These include open patterns and relative clause configurations. Ellipsis-related phrasals offer the same benefits as multi-word expressions recorded in the lexicon: they permit a complex entity to be analyzed as a whole, thus bypassing the need for compositional analysis. The processing of each phrasal pattern involves (a) detecting whether an input text matches the pattern, and if, so, (b) selecting the verbal head of the sponsor. Decisions about which non-head elements should be included in the sponsor are made later.

Below are the open patterns currently included in ViPER.⁵ They fire in the order listed. NP indicates a noun phrase, AUX indicates the auxiliary verb licensing the ellipsis, < > indicates one or more

missing, suggesting ellipsis. Although implementing this generalization would not be completely straightforward, we believe that this would be a fruitful direction for further investigation.

⁵We developed ViPER before reading Bos and Spenader 2011, which delineates an inventory of so-called “source-target patterns” that are similar in spirit to our elliptical phrasals.

related strings recorded in lists,⁶ () indicates an optional element, and * indicates any inflectional form of a listed verb.

1. <all> (NP) (that) NP (adv) AUX

(12) Larry *did* everything and anything he possibly could [e].

2. as <much> (NP) (as) NP AUX

(13) “He is in excellent form and he *knows the Basin Reserve* as only a hometown player can [e],” ...

3. <whatever> NP AUX

(14) Locals have also torched homes to settle old scores, while gangs of toughs have been taking advantage of the situation to *loot* whatever they could [e].

4. as no one else AUX

(15) Moleon gained recognition, *developing* sources as no one else could [e].

5. do* <all> NP can*

(16) “We *did* all we could [e] ...”

6. (in) <ways> NP AUX

(17) Investors who engage in late trading can take advantage of events that occurred after the markets closed and *profit* in ways other shareholders can’t [e].

7. more (adv) than NP (adv) <thought> NP (adv) AUX

(18) It’s worth it in the end, because I’ve *done* more than I ever imagined I could [e].

8. that NP AUX

(19) “I did not see Newcastle’s Titus Bramble and Jonathan Woodgate *make* the mistakes that our defenders did [e].

9. <the way> NP <knew> NP (adv) AUX

(20) Georgia (10-2) was fourth and idle coming into the final weekend – behind Missouri, West Virginia and Ohio State – but didn’t automatically *rise* two spots the way coach Mark Richt thought they should [e].

⁶For example, <all> includes strings like *everything* and *anything and everthing*, and <much> includes strings like *many*, *few*, *little*, etc.

10. *If*-constructions in contexts like (21), in which the elided VP has a postcedent rather than an antecedent.

(21) The message was that if we want to [e], we can *come down out of the hills*.

ViPER's three relative clause patterns are detected using dependencies generated by the CoreNLP parser. For example, if the parse contains the dependencies *rcmod*(*X, ellipsis-licensor*) and *dobj*(*Y, X*), then Y is selected as the head of the sponsor, as in (22).⁷

(22) We are just here to *play* the best cricket we can [e] and progress from there.
rcmod(cricket-9, can-11)
dobj(play-6, cricket-9)

Elliptical contexts that do not match phrasal patterns are passed on to the next stage of processing.

3.2 Simple Parallel Configuration Detector

Syntactically simple contexts that show specific types of structural parallelism provide high predictive power for ellipsis resolution without the need for deep semantic or pragmatic analysis. We will define what we mean by structural parallelism and syntactic simplicity in turn.

Structural Parallelism. Each structurally parallel context contains an ellipsis clause directly preceded by a conjunct that is syntactically connected to in one of three ways that can be loosely described as exhibiting syntactic parallelism: coordination (be it of clauses or verb phrases), parataxis, and certain types of main/adverbial clause pairs. The CoreNLP parser calls these dependencies **conj** (23), **parataxis** (24) and **advcl** (25), respectively.

- (23) They *kept attacking* and we didn't [e].
conj(kept-2, did-6)
- (24) In one walk, they *used poles*; in the other, they didn't [e].
parataxis(used-6, did-14)
- (25) Oakland's Barry Zito took the victory, *hurling eight innings* as Santana did [e].
advcl(hurling-9, did-14)

The motivation for exploiting syntactic parallelism to predict ellipsis resolution derives both from the well-documented linguistic effects of parallelism (e.g., Goodall 2009, Hobbs and Kehler 1998, McShane 2000) and from the successful results of our experiments.

⁷Here and hereafter, we list only those dependencies relevant to the given point.

Syntactic simplicity. The predictive power of parallel configurations decreases significantly if the conjuncts – particularly the first – contain relative or subordinate clauses because such constituents provide additional candidate sponsors for the elided verb phrase. For example, if we rewrite corpus-attested (26) such that the first clause includes several embedded clauses, as in (27), it becomes necessary to carry out sophisticated reasoning about the world to determine which preceding verb is the leftmost constituent of the sponsor: *autographed? broken? playing? left?*

- (26) Bush *autographed a steady stream of memorabilia*, but Garciaparra, his hand in a cast, could not [e].
- (27) Bush *autographed a steady stream of memorabilia*, but Garciaparra, who had broken his wrist playing soccer the day before he left for the conference, could not [e].*Invented*

We operationalized the notion of Simple Parallel Configuration in terms of the output of the CoreNLP parser. Configurations are deemed Simple Parallel if they contain:⁸

1. exactly one instance of what we call a *whitelisted dependency*: a conj, advcl or parataxis dependency that links the modal element licensing the ellipsis with an element from the sponsor clause. Note that conj dependencies that take non-verbal arguments are ignored since they can indicate, for example, nominal conjunction structures: *Lulu and Fido*
2. no instances of a *blacklisted dependency* – i.e., a ccomp, rmod, dep or complm dependency, which indicate various types of embedded verbal structures that complicate matters by offering competing candidate sponsors
3. one or more instances of a *gray-listed dependency*, defined as an xcomp or aux dependency that takes as its arguments matrix and/or main verbs from the sponsor clause.

For example, the parse for (28) includes one whitelisted dependency and three graylisted dependencies.

- (28) John wanted to *try to start to juggle* and did [e].*Invented*
 Whitelisted: **conj**(wanted-2, did-10)
 Graylisted: **xcomp**(wanted-2, try-4), **xcomp**(try-4, start-6),
xcomp(start-6, juggle-8).

⁸Definitions of the dependencies can be found in CoreNLP dependencies manual, available at http://nlp.stanford.edu/software/dependencies_manual.pdf.

Although graylisted dependencies make sponsor clauses more complicated, and although contexts containing them can offer more than one reasonable resolution for the ellipsis (in (28), either *try* or *start* could be interpreted as the left-hand constituent of the sponsor), the added complications can be managed to a large degree by the rule sets for treating sponsor-clause matrix verbs described in Appendix B.

We configured six search patterns to detect the sponsor clause for VP ellipsis in Simple Parallel Configurations.⁹ The first three patterns do not permit modal or other matrix verbs in the sponsor clause (NoMod), whereas the second three do (Mod).

1. Simple Conjunction NoMod

- (29) The audience *loved it*, but one critic didn't [e].
Whitelisted: **conj**(loved-3, did-9)

2. Simple Parataxis NoMod

- (30) Brihmat's four siblings *became French*; he did not [e].
Whitelisted: **parataxis**(became-5, did-9)

3. Simple Adverbial Clause NoMod

- (31) *Stop us* if you can [e].
Whitelisted: **advcl**(stop-1, can-5)

4. Simple Conjunction Mod

- (32) Fans in other countries could *apply on the Internet*, but fans in the host nation could not [e].
Whitelisted: **conj**(apply-6, could-17).
Graylisted: **aux**(apply-6, could-5)

5. Simple Parataxis Mod

- (33) We can *talk ourselves into worse outcomes*; of course we can [e].
Whitelisted: **parataxis**(talk-3, can-12)
Graylisted: **aux**(talk-3, can-2)

6. Simple Adverbial Clause Mod

- (34) I tried to *drive* hard as I could [e].¹⁰
Whitelisted: **advcl**(drive-4, could-8)
Graylisted: **xcomp**(tried-2, drive-4)

⁹Remember, detecting the precise sponsor within that clause – e.g., deciding whether or not to include modal verbs – is undertaken later.

¹⁰This example was not treated by the phrasal detector because our pattern includes a leading *as* (as ADV as NP can*).

Simple Parallel Configurations do not represent a large percentage of elliptical examples in the corpus. However, many complex sentences can be rendered functionally simple through automatic syntactic tree trimming, a process described in Appendix B. The results of tree trimming are shown in example (6), repeated here as (35); the trimmed portion is enclosed in curly brackets.

- (35) “{We’re celebrating the fact that we’re living in a time where,} when we want to *be in the kitchen*, we can [e],” says Tamara Cohen, Ma’yan program director.

Apart from tree trimming, another method of subsuming more contexts under the definition of Simple Parallel Configuration is by merging single-clause elliptical sentences with the preceding sentence. ViPER does this by replacing the period with a semi-colon, which forces a parataxis dependency among the clauses, as shown by the contrast between (36) and (37).¹¹

- (36) A treaty would *require US Senate ratification*. An executive agreement would not [e].
- (37) A treaty would *require US Senate ratification*; an executive agreement would not [e].

In the evaluation, we include this single merging rule under the rubric of tree trimming since it fulfills the same purpose: creating a treatable Simple Parallel Configuration from an input that does not, in its raw form, conform to the necessary requirements.

This concludes our discussion of ViPER’s methods of exploiting Simple Parallel Configurations to detect treatable instances of VP ellipsis and detect which clause contains their sponsor. Now we turn to a completely different strategy for treating additional examples that involves seeking correlations between modal elements, no matter the syntax.

3.3 The Walkback Strategy for Paired Modalities

The intuition behind the Walkback Strategy for Paired Modalities is that some modalities are semantically paired – *could/couldn’t*, *tried/failed*, *wanted to/couldn’t*, *did/didn’t*, etc. So, if a pair is used in close succession, and if the second lacks an overt VP complement, then it is likely that the scope of the first should resolve the ellipsis of the second, as in (38).

- (38) “That caused an outcry, imagine what it will be like if he wants to *appear at his own trial* and can’t [e].”

¹¹The merged version with the semi-colon was presented to annotators.

To exploit this observation, ViPER walks back through the text in search of a modal verb that is semantically paired with the ellipsis-licensing modal. If the first verbal structure it encounters contains such a pair, then the clause with the paired modal is considered the sponsor clause. ViPER does not skip over verbal structures in search of a paired modal since a skipped-over verb could be the actual sponsor.

3.4 Recapping

We have just presented three strategies for selecting the sponsor clause for an elided VP: the use of phrasal patterns, the Simple Parallel Configuration, and the Walkback Strategy for Paired Modalities. All of these detect a treatable subest of elliptical contexts and identify the clause that contains the sponsor. What remains to be done is deciding which constituents in the sponsor clause should be included in the ellipsis reconstruction. We call this “selecting the actual sponsor.”

4 Selecting the Actual Sponsor

ViPER selects the actual sponsor from the already-detected sponsor clause by (a) determining whether matrix verbs (including modals) should be included in, or excluded from, the sponsor, and (b) determining whether internal arguments and adjuncts of the verbal head should be included in, or excluded from, the sponsor. We consider each of these subtasks in turn.

Matrix verbs in the sponsor clause – such as modals and aspectuals – can be included in, or excluded from, the ellipsis reconstruction. ViPER’s analysis of modal verbs is rather involved, as detailed in Appendix C. In brief, if the modal verbs in the sponsor-clause and ellipsis-clause are semantically paired – as defined by pre-recorded lists – the sponsor-clause modal verb is *excluded* from the ellipsis resolution; otherwise, it is *included*. By contrast, ViPER’s treatment of non-modal matrix verbs is simple and straightforward, albeit not yet sufficiently sophisticated to guarantee the correct answer. If a non-modal matrix verb is the outermost verb in the structure, it is *excluded* from the sponsor (in (39) *refused* is excluded). By contrast, if a matrix verb is, itself, scoped over by another matrix verb, it is *included* in the sponsor (in (40) *asked* is excluded but *stop* is included).

- (39) Mylonas refused to *reveal the ransom paid for his release*, although police sources did [e].
- (40) The charity had asked the school and several organizations to *stop using the name*, and they did [e], said Sister Nirmala.

ViPER determines whether or not to include internal arguments and

adjuncts in the ellipsis resolution as follows. All terminals subsumed by the sponsoring head in the parse tree are included in the resolution unless the elided category itself is among those terminals; in that case, the verbal head alone serves as the sponsor. This strategy is suboptimal on two counts. First, it is too simplistic, as we have not yet deeply studied this aspect of the problem. Second, it is completely dependent upon the parser’s treatment of the given argument or adjunct. In some cases, an argument or adjunct that should participate in the resolution is parsed as a sibling rather than a child of the verbal head. In such cases, the best ViPER can do is achieve “head correct” resolution status. Clearly, this is a fertile area for further study.

There are, of course, many other challenges in selecting the actual sponsor for elided VPs, as detailed in the evaluation below.

4.1 Pre-Evaluation Wrap Up

This concludes the description of the current state of ViPER. Its output is (a) a set of examples that it believes include VP ellipsis prior to a hard discourse break, a soft discourse break, or a conjunction; (b) the subset of examples that it believes it can treat with reasonable confidence; (c) ellipsis resolutions for the latter; and (d) metadata indicating which resolution strategy was applied to each example, for purposes of testing and debugging both the implementation and the knowledge supporting it.

5 Evaluation

ViPER is a knowledge-based system that does not require an annotated corpus for training. In fact, we used human annotations only to evaluate whether ViPER had correctly resolved those instances of VP ellipsis that it chose to treat. By contrast, a full-scale annotation effort, like those used for the well-known system competitions, would have required creating a formal task definition, training annotators to follow it, and overseeing the work of those annotators – a formidable program of work that we did not undertake. Instead, our goals in evaluating ViPER were (a) to have some measure of current system performance as a benchmark for future advances, and (b) to identify the specific challenges posed by this problem space. In this discussion, we will motivate all aspects of our rather novel evaluation suite.

Evaluation was carried out on the Gigaword corpus, the same corpus used for linguistic investigation and algorithm development. Although we did not initially divide the corpus into testing and evaluation portions, we believe that the evaluation is a fair representation of system functioning since (a) examples for the different stages of work were

selected at random from this very large corpus, (b) we carried out no special tuning of our programs to cover specific examples, and (c) ViPER is not a statistically trained system. The evaluation proceeded according to the steps defined by the following subsections.

5.1 Extracting Candidate Examples from the Corpus

ViPER searched the Gigaword corpus for potentially elliptical sentences containing any of a list of auxiliaries followed by a hard discourse break (HDB), a soft discourse break (SDB), or a listed conjunction. This yielded 26,988 unique instances, with relative frequencies of 18%, 20%, and 62%, respectively. ViPER then pruned out false-positive examples involving homonymous nominals, parentheticals, and certain idiomatic nonelliptical expressions such as *cannot but* (“We could not but resume”). This left 18,027 candidate examples.

5.2 Selecting Treatable Examples

In order to exclude superfluous detail from the evaluation results, we grouped the ellipsis resolution strategies into the following eight, shown in boldface:

- **Phrasals**: Listed open patterns, including the *if* + postcedent pattern
- **RcMod**: Elliptical relative clause constructions
- Simple Parallel Constructions, which could contain or exclude matrix verbs in the sponsor clause (Mod, NoMod), and could leverage or not leverage syntactic trimming (Trim, NoTrim): **SP-Mod-NoTrim**, **SP-Mod-Trim**, **SP-NoMod-NoTrim**, **SP-NoMod-Trim**
- The Walkback Strategy, which could include or exclude syntactic trimming (Trim, NoTrim): **Walkback-NoTrim**, **Walkback-Trim**.

ViPER selected treatable examples by applying these strategies in sequence. For evaluation, it randomly selected 55 examples per configuration, from which we removed duplicates.¹²

5.3 Annotating the Examples Treated by ViPER

Two undergraduate students and one graduate student, all of whom had taken at least one course in linguistics, served as annotators. We presented them with the examples that ViPER had selected but without any indication of the elided category or the sponsor that ViPER had chosen. The examples were presented in separate groups, corresponding to the groups above, but without any indication of group

¹²The Walkback-Trim configuration had only 13 hits.

membership.¹³ Annotators were provided with the following instructions, excluding organizational details:¹⁴

- Indicate instances of VP ellipsis with [e].
- If the example contains no VP ellipsis, insert [NE] (not elliptical) before the example.
- If there is a unique textual sponsor, surround it by square brackets.
- If the sponsor is outside of the provided context, type [NS] (no sponsor).
- If there are multiple possible sponsors, type [Mult] before the example and select all possible sponsors using multiple pairs of square brackets.
- If the example is problematic in some way, type [Prob] before the example. This is for examples that are incomprehensible or seriously misformed.
- If there are surface differences in the form of the sponsor and the elided element, such as differences in tense or number, ignore them.
- If there is “sloppy coreference” of internal arguments, go ahead and choose the sloppy coreferent but type [Sloppy] before the example. For example: [*Sloppy*] *I [rode my bike] but he didn't [e].*
- Do not select discontinuous constituents. For example, do not include [cabbage] in the following resolution: *You should [eat] as much cabbage as you can [e].*
- Include in the sponsor any objects, modifiers, etc., that are understood to be part of the recovered meaning as long as they are contiguous with the verbal head: *He wanted to [play in the morning] but couldn't [e].*
- Keep in mind that elided slots can have postcedents, not only antecedents: *If you can [e], [pick up a bottle of wine on the way home].*

5.4 Measuring Interannotator Agreement

Given the underspecification of our annotation conventions, the known insufficiencies of contiguous string selection as a method of indicating the sponsors for elided VPs, and the availability of multiple possible sponsors in some contexts, measures of interannotator agreement are not very informative. Moreover, in this work we did not undertake to

¹³We did not think that randomizing example order across implemented resolution strategies would affect the annotators' decisions.

¹⁴These annotators had participated in a penultimate evaluation as well, which involved our fielding some questions.

advance the state of the art in annotation methodologies or measurements. So, we limited the measure of interannotator agreement to exact matches on sponsor selection using Krippendorff’s alpha coefficient (Krippendorff 2010), with a result 0.877.¹⁵

5.5 Analyzing Discrepancies in Annotations

Below we present an initial classification of interannotator discrepancies, with an exhaustive analysis meriting a separate paper. We do not include counts for each class because some class descriptions overlap and some examples fall into multiple classes. Eventualities fall into two superclasses: annotation errors, and instances in which multiple annotations are reasonable.

Annotation Errors. Annotation errors included typos (e.g., mistyping a bracket), failing to include applicable metadata labels, and failing to interpret the context correctly. An example of the latter occurred in (41), in which one annotator excluded *and the West Bank* from the sponsor.¹⁶

- (41) Jordan “insists that these displaced can [return to Gaza and the West Bank] if they want to [e].”

Multiple Reasonable Options. Below is a sampling of contexts in which more than one annotation is correct.

- When [e] is the third member of a chain of coreference, annotators could choose either the first or the second member as the sponsor. Two of the annotations for (42) are shown using different bracketing structures.

- (42) However, Beijing still [rules the country with harsh authoritarian methods] in the provinces and will [continue to do so] for as long as it can [e].¹⁷

- Some internal arguments or adjuncts can be included in, or excluded from, the sponsor. In many cases, the difference involves being more complete semantically vs. sounding better.

- (43) Nuclear power may [[give] NASA’s long-range missions] the speed and range that combustion engines can not [e], but research is sputtering for lack of funds.

¹⁵See Artstein and Poesio 2008 for a discussion of interannotator agreement.

¹⁶When discussing the evaluation, we present examples using the same conventions as annotators used: i.e., square brackets indicate the sponsor.

¹⁷It is not clear why the annotator excluded *in the provinces* from the first sponsor option.

- A longer and a shorter text span can both be possible, with the variants representing something other than the inclusion/exclusion of internal arguments and adjuncts.

(44) “I didn’t get to [go out and [play]] the way I could [e],” James said.

(45) Dat You Miz Blue spotted the field several lengths at the break, then [recovered and [won the Belmont Park feature Wednesday]] just as a 2-5 shot should [e].

- Annotators could recognize or not recognize the availability of multiple answers.

(46) [Mult] “We had to [go out and [play the game]] just like they did [e].”

- A different treatment of modality could account for differences in annotations. For example, in (47) two annotators selected [do] as the sponsor, whereas the other selected [try]¹⁸.

(47) “I don’t know that I can satisfy all the privacy advocates,” she said, “but I want to [try] to [do] everything I can [e].”

- A slogan or title can be considered either elliptical, with a textually unavailable sponsor, or non-elliptical.

(48) “Brothels fascinated Greene,” notes the biographer, who reproduces a list Greene compiled of his 47 favorite prostitutes, awarded nicknames including “Russian Boots” and “the one who wouldn’t [e].”

Either way, ViPER will get such examples wrong: under the elliptical analysis, there is no textual sponsor; and to arrive at a non-elliptical analysis, ViPER would need to detect the idiomaticity of the collocation.

- Some surface aspect of the text can make it difficult to select a sponsor, especially given the ground rule of not selecting non-contiguous text spans. For (49), two annotators selected *pull* (despite the lack of the associated particle *off*), while the other selected *pull that shot off* with the metadata [Sloppy].

(49) “I feel I can [[pull] that shot off]; that’s just one of those I didn’t [e].”

- Many other difficult annotation issues occur rarely or as singletons. For example, (50) could be considered non-elliptical with fronting

¹⁸It is unclear why the latter annotator did not include *to do* in the resolution – i.e., *try to do*.

of the VP for emphasis, or it could be considered elliptical in the normal way.

(50) “And [do it] we must [e].”

Finally, in (51), the sponsor itself includes an elided VP which must resolve to the earlier instance of *disrupt you* – an eventuality not covered by our annotation conventions. Two of three annotators selected just *let them*, and we considered this a sufficiently correct answer for the gold standard.

(51) “They can disrupt you if you [let them], and we didn’t [e].”

We will refrain from presenting more examples of this type, interesting as they are from the linguistic perspective. Suffice it to say that it would take a very detailed, difficult-to-master set of annotation rules to cover the many phenomena that complicate the text-level selection of a sponsor for elided VPs.

5.6 Creating the Gold Standard

For 81% of examples (320 out of 393), all annotators chose the same annotation: it was counted as the gold standard with no quality checking by the authors and no exploration of cases in which the annotators were forced to select a non-optimal resolution. For example, in (52) all annotators selected *afford*.

(52) We can [afford] a war in Iraq, but when it comes to a massive disaster, we can’t [e].”

If the system were to select the same sponsor, we would have no reason to look at this example or to note the imprecise nature of the resolution – which would ideally be *afford it [= a massive disaster]*.

For the remaining 73 examples, there were multiple annotations. The authors – working in part separately (in an attempt to achieve objectivity) and in part jointly (to arrive at a gold standard in finite time) – made final decisions about which variants to include in the gold standard. Let us mention a few policies, as well as a sampling of the many judgment calls, that contributed to the creation of the gold standard.

One policy was to exclude sponsors that would lead to recursion: e.g., in (53) *painting the picture* was rejected as a sponsor.

(53) They [painted] the picture they wanted to [e], and these people reacted.

Another policy was to rather liberally accept “close but not quite” sponsor selections in examples that did not offer any good, contiguous text-level options. For example, in (54) we accepted either *have* or *done* as sponsors.

- (54) “We [have]n’t [done] nearly as much in that field as we should [e].”¹⁹

We did not, however, include in the gold standard the complex annotation *haven’t done* plus the metadata [Sloppy]. Although this option was a reasonable way out of an impossible situation, including it would have given ViPER credit for an imprecise resolution since the system does not have the option of appending a [Sloppy] label to a sponsor selection (nor is negation removal typically subsumed under sloppy coreference).

In at least one case, (55), the sponsor is a non-contiguous set of propositions – an expressive means we did not offer to annotators.

- (55) [Mult] “There are some who want to [intimidate the Iraqi people], who want to [intimidate the coalition], they want to [intimidate the international community] and they can not [e].”

Two annotators selected the three-part annotation shown here, presumably intending to extend the definition of [Mult] to include the union of selections. ViPER could, at best, receive partial credit for such examples since it did not attempt dynamic set concatenation.

A number of contexts contained more than one elided VP. The most complete annotation would cover all ellipses, as shown in (56).

- (56) [2e] [NS] “We are very sad we did [e] but we must understand what football is, they [scored] and we didn’t [e].”

However, if an annotator treated only one of the instances, we included that in the gold standard as well since, in selecting the example, ViPER could have been focusing on one or both instances.

A particularly interesting case is illustrated by (57), for which one annotator selected the NP *everything* as the sponsor – a perfectly reasonable response, but one that resolves an elided VP with a nominal.

- (57) “They [did [everything]] better than we did [e].”

The linguistically more appropriate sponsor is *did everything* (which would be read in the ellipsis reconstruction as “we did do everything”), since English does not permit direct object ellipsis.

¹⁹Since the texts presented to annotators were tokenized – and words like *haven’t* were separated into *have* and *n’t* – annotators could readily select non-negated elements, like *have*, as the sponsor.

To summarize, when different annotations were provided they could be functionally equivalent (differing only in metadata tags), they could all be correct, or one or more could be incorrect. The authors considered up to three sponsors to be sufficiently correct to be included in the gold standard for each example.

5.7 Evaluating the System Against the Gold Standard

We judged ViPER’s ellipsis resolutions as being Exactly Correct, deserving Partial Credit, or being Incorrect. **Partial Credit** covers the following eventualities:

- **Head correct:** The head alone – and, optionally, a verbal particle – is selected and is correct, but other constituents should have been selected as well.
- **Extra Incorrect:** The head is correct but some of the extra constituents ViPER selected were not correct (there could be too many or too few).
- **Modality Error:** ViPER incorrectly either included or excluded a modal verb in the sponsor. If it incorrectly selected a modal verb but failed to include the main verb in the sponsor as well, the resolution was marked Incorrect. This class only refers to modalities. If the treatment of non-modal matrix verbs was wrong, the resolution was considered Incorrect.
- **Conjunction Partial Credit:** The sponsor contains a conjunction structure and ViPER selected some correct portion of that structure – usually either the head of the first conjunct or the whole first conjunct.

Incorrect covers the following eventualities:

- **NE:** The context is not elliptical but ViPER thought it was.
- **NS:** The sponsor is not available in the context but ViPER did select a sponsor.
- **Other:** This category includes both outright mistakes and cases that actually might deserve partial credit, but in order to recognize that, one would have to fundamentally understand both the parser output and the way ViPER uses it. For example, in certain types of examples containing copulas or auxiliaries, such as (58), ViPER regularly selects a non-verbal head as the sponsor.

- (58) “I thought that today was very hot weather conditions, and I was very surprised that we were not [treated] as we should [e],” she said.

TABLE 1 Overall results of the system evaluation, divided into completely correct (Correct), partial credit (Partial), and incorrect (Incorrect).

Strategy	Evaluated	Correct	Partial	Incorrect
Phrasals	55	32 (58%)	18 (33%)	5 (9%)
RcMod	55	32 (58%)	9 (16%)	14 (25%)
SP-Mod-NoTrim	55	45 (82%)	6 (11%)	4 (7%)
SP-Mod-Trim	55	38 (69%)	10 (18%)	7 (13%)
SP-NoMod-NoTrim	55	37 (67%)	7 (13%)	11 (20%)
SP-NoMod-Trim	52	26 (50%)	5 (10%)	21 (40%)
Walkback-noTrim	53	25 (47%)	15 (28%)	12 (23%)
Walkback-Trim	13	5 (38%)	5 (38%)	3 (23%)
TOTAL	393	239 (61%)	75 (19%)	80 (20%)

TABLE 2 A breakdown of Partial Credit outcomes. Eventualities are Head correct (Head), Conjunction partial credit (Conj.), Extra incorrect (Extra) and Modality error (Modality).

Strategy	Head	Conj.	Extra	Modality
Phrasals	9	0	9	0
RcMod	7	1	1	0
SP-Mod-NoTrim	2	0	3	1
SP-Mod-Trim	3	0	3	4
SP-NoMod-NoTrim	3	0	4	0
SP-NoMod-Trim	1	0	4	0
Walkback-noTrim	5	4	5	1
Walkback-Trim	0	2	3	0
TOTAL	30	7	32	6

It would not be difficult to create a string-level fix-up rule to reinsert the associated verb, we simply have not done so yet and penalized the system for that. Note that in this example, annotators will also struggle, since the sponsor, *be treated*, is not available as a contiguous text string.

Table 1 shows the overall classification of examples into Completely Correct, Partial Credit, and Incorrect. Table 2 breaks down Partial Credit outcomes into the relevant subclasses, and Table 3 does the same for Incorrect outcomes.

5.8 Exploring Errors

As expected, the use of syntactic tree trimming tended to increase error rates, and our overly simplistic rules regarding which non-head material to include in the sponsor led to many instances of partial, rather than full, credit. Both of these underdeveloped modules would benefit from

TABLE 3 A breakdown of Incorrect outcomes into false positive ellipsis detection (NotElliptical), elliptical examples with no sponsor in the provided context (NoTextSponsor), and all other mistakes (OtherError).

Strategy	NotElliptical	NoTextSponsor	OtherError
Phrasals	0	1	4
RcMod	3	5	6
SP-Mod-NoTrim	2	0	2
SP-Mod-Trim	0	0	7
SP-NoMod-NoTrim	8	1	2
SP-NoMod-Trim	4	1	16
Walkback-noTrim	2	3	7
Walkback-Trim	0	1	2
TOTAL	19	12	49

further study of the corpus and refinement of the treatment methods.

Our evaluation standards were rather rigorous. For example, we did not give the system partial credit if there was more than one deduction. In (59) even though the system’s selection was in the correct clause, the incorrect inclusion of modality coupled with the exclusion of the internal argument led us to judge it as Incorrect.

- (59) We want to be [able] to communicate with him, but for the time being we can not [e].

Similarly, in (60) the best answer is *made them*, whereas ViPER selected the penultimate instance of *made* with no object; therefore, it received no credit.

- (60) “We have been in litigation long enough to know that sometimes you get something and sometimes you don’t ... we made our arguments and [made them] the best we could [e].”

And in (61), although ViPER selected the correct string – the first instance of *return* – the best answer is the second instance of *return*, so the resolution was marked Incorrect.

- (61) But no one really thinks Prague will let Germans return en masse and few would [return] even if they could [e].

If the input was grammatically corrupt but still understandable by people, the example remained in the corpus. In (62), *received* includes a typo (an extra *d*), making ViPER think there was an elliptical gap after *would*.

- (62) [NE] Tourism Bureau officials said that in addition to Konrath, the five arrivals before and after him would also received a fine glass artwork.

We expanded the gold standard in a few cases during system evaluation. For example, none of the annotators provided the arguably best answer for (63), which is [NS], since the resolution should be *do*.

- (63) “Finance Minister Chidambaram can deliver this promise only if he mobilises additional revenues and reduces wasteful expenditure, both of which he must [e],” it said.

Even though ViPER was not tasked with detecting instances of sloppy coreference, we asked annotators to indicate them in order to measure the prevalence of this phenomenon. The tag was used for eight examples. In three of them, it was used in a way not intended by the guidelines. Of the five correct usages (1.3% of examples), twice one annotator used the label, twice two annotators used it, and once all three used it. The authors did not review the corpus in search of missed instances of sloppy coreference. As concerns the related issue of type vs. instance coreference of the elided verb and its sponsor, this was not treated either by the annotators or by the system.

6 Related Work

Reference and ellipsis have been treated in different ways within several fields. Descriptive linguists, theoretical linguists, philosophers, and computational linguists wearing their theoretical hats have offered theories that are difficult to implement, typically because they require prerequisites that remain beyond the state of the art. For example: Hardt’s (Hardt 1997) system for treating VP ellipsis requires as input a perfect, manually corrected, parse; Webber’s (Webber 1988) theory of pronominal coreference requires full knowledge of discourse structure; the Centering Theory approach to pronominal coreference (Grosz et al. 1995) is difficult to operationalize due to problems in defining key concepts (Poesio et al. 2004); and virtually all work on strict vs. sloppy anaphora has been carried out in a theoretical paradigm that does not endeavor to render its findings computer-tractable.

Another source of work on reference – and less so, ellipsis – is mainstream NLP, which has treated a subset of overt referring expressions quite extensively. This subset is defined, details aside, as expressions whose sponsor is an overt, contiguous, coreferential NP. The MUC coreference task definition (Hirshman and Chinchor 1997), which details these guidelines, has been very influential in guiding corpus annotation and subsequent system development in this domain for almost two decades. Many machine learning methods, particularly supervised ones (see Olsson 2004 for an overview of methods brought to bear), have been applied to this task. But even state of the art systems like

CoreNLP's reference resolver (Lee et al. 2013) are struggling with some types of referring expressions, such as personal pronouns. As Lee et al. (ibid) write: "The main conclusion of our error analysis ... is that the plurality of our errors are due to shallow knowledge of semantics and discourse. This result points to the crucial need for more sophisticated methods of incorporating semantic and discourse knowledge" (Lee et al. 2013). Other high-level analyses of the state of the art in reference resolution can be found in Mitkov 2001 and Stoyanov et al. 2009, who share the opinion that the commonly adopted evaluation metrics lead to artificially inflated perceptions about where we stand in automating the process of anaphora resolution. Stoyanov et al. explain, "... [T]he assumptions adopted in some evaluations dramatically simplify the resolution task, rendering it an unrealistic surrogate for the original problem" (Stoyanov et al. 2009).

One of our goals in developing ViPER was to not simplify the problem in advance but, rather, to task the system with automatically detecting which instances it could treat. ViPER's focus on treating simpler instances of a difficult phenomenon resonates with Baldwin's CogNIAC system (Baldwin 1997).

As concerns the supporting functionality of syntactic trimming, much of the past work has been in service of text summarization. For example, Knight and Marcu 2002 use a methodology that involves aligning sentences from a source document (called 'Text') with sentences from manually generated abstracts of the document (called 'Abstract'), then using these <Abstract, Text> tuples to learn how to trim Texts into Abstracts. Gagnon and Da Sylva 2005 trim sentences based on a dependency parse, removing subtrees that represent certain types of relations, such as prepositional complements of the verb, subordinate clauses and noun appositions. Perhaps the closest precedent to our approach is the one reported in Vanderwende et al. 2007, which applies five linguistically-informed trimming patterns. Apart from summarization, sentence trimming has been applied to headline generation, event extraction, and subtitle generation.

We did not use previously annotated corpora for system development or evaluation for two reasons. First, we knew that our methods would have relatively low recall; therefore, typical pre-annotation efforts would not have yielded a sufficient number of examples to test all of our linguistic hypotheses. Second, using externally-prepared resources requires overhead. For example, Bos and Spenader 2011 remark about the annotations prepared by Nielsen 2005, "... for at least the WSJ files, the annotation is not easily reusable because it has been carried out on tokenized and parsed data, rather than on the texts in the form they

are distributed by the LDC. This means that reusing the WSJ annotations requires re-identifying each example in the raw texts” (p. 467). Our approach to annotation was the fastest and most efficient way to support development and evaluation of ViPER. We would be happy to share any of our resources for research purposes.

7 The Big Picture

A natural question is, how frequent is VP ellipsis? We cannot answer that because there are no foolproof methods of automatically detecting elided categories, and counting manually would be impractical. But no matter the frequency of VP ellipsis, we believe that ViPER – or another implementation of its approach – could be useful in two ways. First, if applied to large corpora, it could treat at least some instances of VP ellipsis, which is better than treating none. Second, ellipsis will be encountered by intelligent agents in all types of cognitive systems, such as our research group’s OntoAgent cognitive architecture (McShane and Nirenburg 2012). In fact, we are currently incorporating ViPER into the OntoSem2 incremental semantic analysis system that provides language analysis capabilities to OntoAgents. ViPER’s output serves as heuristic evidence for full semantic reconstruction of elided VPs. By year’s end we expect to have reportable results of semantic VP-ellipsis resolution in OntoSem2.

In seeking approaches to treating truly difficult linguistic phenomena like reference and ellipsis, or the semantic analysis ultimately needed to inform them, we directly confront one of the biggest challenges of artificial intelligence: delineating a program of research and development that will result in fundamental solutions to hard problems while also offering useful, short-term results. We believe that it is essential that important problems of language analysis remain in purview even if currently preferred methods of system development cannot conveniently treat them. In fact, not only is there a place for goal-driven problem selection alongside the currently more prevalent method-driven strategy, the goal-driven approach highlights some of the most compelling problems posed by natural language, problems that will need to be solved as we move toward configuring truly intelligent artificial agents.

Appendix A: Parenthetical Detection

To develop a parenthetical expression detector, we first extracted from the Gigaword corpus all contexts in which a verb from our list of auxiliaries (including modal auxiliaries) was followed by a soft discourse break: a comma, dash or open parenthesis. We then manually classified a subset of examples into three categories:

- **Elliptical**: “auxiliary + comma” signals VP ellipsis.
- **Not elliptical**: “auxiliary + comma” introduces a parenthetical, which is followed by the complement of the auxiliary
- **Not applicable**: either
 - the context is corrupt or unintelligible, or
 - the auxiliary word is not a verb to begin with (we did not tag the candidates for part of speech prior to this analysis).

The classification was carried out by one lightly trained annotator who was directed to work quickly since the results were to be used only to guide system development, not as a formal gold standard.

For the non-elliptical (parenthetical) contexts, we manually classified examples based on the syntactic structure of the parenthetical portion, resulting in the 12 classes listed below. In classes 1-11, the auxiliary precedes the parenthetical; in class 12, two modal verbs share a complement, which follows the second one. For each class, we present the pattern followed by a sampling of strings that match it.

1. CoreNLP’s *prn* constituent, which detects various types of parentheticals directly: *, they wondered,*
2. Conjunction (+ subject) + auxiliary: *and did, and need not, or wishes to, and one should not*
3. Prepositional phrase: *among others, at any price*
4. Adverb: *however, therefore, potentially*
5. Adverbial phrase: *absolutely not, more than ever, God forbid*
6. Conjunction + clause: *as he put it, if it were developed*
7. Special: *as is the case with NP, as is the situation with NP*
8. Conjunction + subjectless past-participial clause: *if untreated, as previously announced, if given in sufficient doses, if needed*
9. Conjunction + adjective (adverb): *if possible, as ADJ as possible*
10. Objectless clause: *it seems, you know, SUBJ feel <believe, imagine, think, guess, hope, etc.>, I am sure*
11. Present participle + object: *gritting our teeth, following a review*
12. Two modals share a scope, both appearing elliptical at the surface:

- (64) “The possibility for events to spiral rapidly out of control in circumstances of darkness, high emotions, low trust and official uncertainty **cannot, and should not, be underestimated**,” DeGolyer said in a report published last July.

The reason for specifying the structure of parentheticals, rather than relying on the pairing of punctuation marks around them, is that the second punctuation mark is often missing, as in (65).

- (65) “The fact that the release has been continued **should not, however** be taken as an indication as to the length of sentence,” presiding judge Richard May warned.

The above classification supported the development of a parenthetical detection program that attempted to prune out false-positive examples from our inventory of potentially elliptical contexts. Parenthetical detection is the final stage of example selection. All examples that remain after this pruning are treated as elliptical.

Appendix B: Sentence Trimming

Sentence trimming – also referred to as sentence pruning, sentence compression, sentence simplification and tree trimming – exploits the fact that some sentence constituents are more central to the meaning of an utterance than others. Constituents of the basic argument structure are core whereas parenthetical clauses, relative clauses, discourse connectives and certain types of modification are not.

We implemented eight trimming procedures that relied on the outputs of CoreNLP preprocessing, basic syntactic analysis, and dependency parsing.²⁰ In the examples below, the portion of the sentence that is trimmed is shown in curly brackets.

1. Strip sentence-initial adverbs. ViPER detects sentence-initial adverbs using a list of over 500 sentence adverbs. The list was created by a combination of introspection and searches using the online version of the COCA corpus (Davies 2008).²¹

- (66) {And then} you’ll start *working at it* as you can [e].

2. Strip Pre-Punctuation Clause. ViPER walks backwards through the text, starting from the elided VP. If it encounters a comma, dash,

²⁰Although some of these procedures are more error-prone than others, we chose not to split the evaluation along yet another axis: aggressive vs. conservative trimming.

²¹For example, we searched for single words, and 2- and 3-word collocations, occurring between a period and a comma.

semicolon or colon, it strips it off along with the preceding context and determines whether the remaining portion is a Simple Parallel Configuration. If so, as in (67), it resolves the ellipsis. If not, as would be the case for the first encountered comma in (68), it continues its march backwards to the next punctuation mark and asks the same question.

(67) {I was OK,} I tried to *find my game* but I couldn't [e].

(68) {I was OK,} I tried to *find my game*, but I couldn't [e].*Invented*

3. Strip matrix verb and preceding context. ViPER walks backwards through the text. If it encounters one of a listed inventory of matrix verbs,²² it removes that verb and all preceding context and evaluates whether the remaining structure is a Simple Parallel Configuration. If it is, ViPER resolves the ellipsis. If not, it continues to walk backwards in search of the next matrix verb, which it evaluates in the same way.

(69) {Barak told Israel TV that} the agents asked if he would *help them in their investigation of the attacks* if he could [e].

4. Strip pre-conjunction material. ViPER walks backwards through the text to the first encountered conjunction. If it is among our listed 28 conjunctions,²³ and if it takes verbal arguments in the dependency structure, then ViPER determines whether the latter conjunct is a Simple Parallel Configuration. If yes, ViPER resolves the ellipsis. If not, it determines if adding another conjunct would result in a Simple Parallel Configuration. For example, when encountering *and* in (70) ViPER will continue on, but when encountering *because* it will stop and strip off the preceding material and resolve the ellipsis.

(70) {My legs make the serve because} you need to *bend your knees* and I couldn't [e].

5. Strip sentence-initial PPs, which are detected from the parse tree.

(71) {In the swimming test,} inosine-treated rats by week eight were able to *properly control their forepaws*, while the untreated rats could not [e].

²²The list used for the reported experiment is: *say, think, add, complain, ask, argue, tell, explain, mean, assume, find, complain, joke, bet.*

²³The list of conjunctions is: *although, and, and that, as, as if, as long as, as though, because, but, even if, even though, how, if, in order that, just like, lest, now that, provided that, since, so, so that, though, unless, until, whenever, whether, whether or not, why.*

6. Strip pre-Sbar *that*. This function strips everything before an instance of *that* functioning as a relativizer or a complementizer, and then it tests if the remainder is a Simple Parallel Configuration. If not, and if there are still too many dependencies, ViPER invokes other stripping methods. If, by contrast, the context ended up being too short, then this method is abandoned for the given sentence. (72) is an example for which this method transformed an overly complex context into one whose VP ellipsis ViPER could resolve.

- (72) {Petra for us has been a symbol of a place that} all of us wanted to *get to* but couldn't [e].

7. Strip parentheticals. We use a slightly modified version of the parenthetical detection algorithm described in Appendix A to strip parenthetical elements.

- (73) Bush *signed a steady stream of memorabilia*, but Garciaparra {- his hand in a cast -} could not [e].

8. Strip non-quotative NP *said/was told*, etc. The collocations *NP said*, *NP was told* and paraphrases thereof, used non-quotatively, are stripped from the main proposition.

- (74) {Belu said} he wanted to *protest*, but {was told} he could not [e].

The stripping rules were applied in the order listed. We did not attempt to optimize their ordering experimentally.

Appendix C: Microtheory of Modality Pairing

Our goal in studying modality pairings was to answer the following questions:

1. If a given pair of modal meanings occurs in a Simple Parallel Configuration, should the modal meaning(s) in the sponsor-clause be included in, or excluded from, the ellipsis reconstruction?
2. Which modal pairs are semantically strong enough to support the Walkback Strategy for ellipsis resolution?

To guide corpus analysis, we created a list of nearly 100 modal correlations and used them for automatic example extraction. The search patterns had the following features:

1. They included groupings of inflectional forms and synonyms: e.g., *try** expanded to all inflectional forms of *try* and *attempt*.
2. They included combinations of modal and aspectual meanings, e.g., *try* to stop VERBing*.

3. They included hypotheses about the ellipsis resolution strategy for each pairing.

The correlations were formulated as patterns like the following:

- doesn't* X ... doesn't* try to [e] ([e] = X)

(75) I didn't *beat her* and I didn't try to [e].*Invented*
- can* start Xing ... can't* [e] ([e] = start Xing)

(76) She can *start running* but he can't [e].*Invented*

Example extraction from a portion of the Gigaword corpus (Graff and Cieri 2003) gave mixed results. Many patterns were not attested at all, particularly those that included both a modal and an aspectual verb: e.g., *can* finish X ... can't**. We excluded those from this version of the microtheory, even if our intuitions about how they would behave in elliptical contexts were strong.

Based on the attested evidence, we configured six hypotheses about the behavior of modal meanings in elliptical contexts. The first four predict whether modal meanings should be retained in, or stripped from the ellipsis reconstruction in Simple Parallel Configurations. The last two suggest which modal correlations are strong enough to support the Walkback Strategy of ellipsis resolution.

Modality Correlation Hypothesis 1. If a Simple Parallel Configuration contains any of the following positive/negative pairs, in either order, then resolve the ellipsis using the scope of the paired modal: can*/can't*, do*/don't*, must*/mustn't*, should*/shouldn't*, want* to/don't* want to, be*/not be*, would*/wouldn't*.

- (77) I can't *change the past* – no one can [e].

The ellipsis reconstruction can include any nested modal or aspectual verbs, as shown by (78).

- (78) Sid wanted to *try to start juggling* but Sophie didn't want to [e].*Invented*

Modality Correlation Hypothesis 2. If a Simple Parallel configuration contains any of the following modal pairs in this order, then resolve the ellipsis using the scope of the sponsor-clause modal: try* to/can*, try* to/do*, try* not to/can, try* not to/do*, can*/want* to, can*/be*, can*/didn't*, can*/should*, must*/can*, must*/can't*, must*/didn't*, should*/didn't*, should*/can*, should*/can't*, can't*/must*, want* to/can*, want* to/can't*.

(79) We can *make it* if we want to [e].

(80) I tried not to *touch the bar*, but I probably did [e].

Modality Correlation Hypothesis 3. This hypothesis includes an extra feature: the correlation between the subjects of the sponsor clause and the ellipsis clause, which can be coreferential or not coreferential. We used CoreNLP coreference processing to make the coreference determinations. The configurations covered by this hypothesis are shown below, with P and Q indicating different subjects, and the ellipsis resolution indicated in square brackets.

- P doesn't* want to X ... P doesn't* [e = X] .
- P doesn't have to X ... P doesn't* [e = X]
- P doesn't want/have/try to X ... Q doesn't* [e = want/have/try to X] .
- P want*/have*/try* to X ... Q doesn't* [e = want/have/try to X]
- P be* trying/intending/starting to X ... Q isn't* [e = trying/intending/starting to X]

(81) People *wanted to give up on him* but I didn't [e].

Modality Correlation Hypothesis 4. If a Simple Parallel Configuration contains any other modal pair, then resolve the ellipsis using the scope of the sponsor-clause modal.

(82) But he would not *predict timing*, as Pelosi did [e].

This ends treatment of paired modalities in the Simple Parallel Configuration. Now we move on to the use of modality matching in the Walkback Strategy.

Walkback Hypothesis 1. If the Walkback Strategy detects any of the following positive/negative pairs, in any order and with any subject correlation (coreferential or not), then resolve the ellipsis using the scope of listed sponsor-clause modality: can*/can't*, do*/don't*, must*/mustn't*, should*/shouldn't*, want* to/don't* want to, be*/not be*, would*/wouldn't*.

(83) On Thursday the Pentagon said Navy SEAL commandos would *accompany the helicopters*, but today it said they would not [e].

Walkback Hypothesis 2. If the Walkback Strategy detects any of the following pairs, in the listed order, with the same subject, resolve the ellipsis using the scope of the listed sponsor-clause modality: try* to/can*, try*to/do*, can*/want* to, can*/be*, can*/didn't*,

can*/should*, must*/can*, must*/can't*, must/didn't*, should*/didn't*, should*/can*, should*/can't*, can't*/must*, want* to/can*, want* to/can't*.

- (84) “If you look at China and Asia and this region here (India), and the Middle East, you can already *play a full schedule without going to the United States* if you want to [e].”

Acknowledgments

This research was supported in part by Grants N00014-09-1-1029 and N00014-16-1-2118 from the U.S. Office of Naval Research. Any opinions or findings expressed in this material are those of the authors and do not necessarily reflect the views of the Office of Naval Research.

References

- Artstein, R. and M. Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics* 34(4):555–596.
- Baldwin, B. 1997. CogNIAC: High precision coreference with limited knowledge and linguistic resources. In *Proceedings of a Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*. Association for Computational Linguistics.
- Bos, J. and J. Spender. 2011. An annotated corpus for the analysis of VP ellipsis. *Language Resources and Evaluation* 45:463–494.
- Davies, M. 2008. The Corpus of Contemporary American English: 450 million words, 1990-present. Available online at <http://corpus.byu.edu/coca/>.
- Fienberg, R. and R. May. 1994. *Indices and Identity*. Cambridge, Mass.: The MIT Press.
- Gagnon, M. and L. Da Sylva. 2005. Text summarization by sentence extraction and syntactic pruning. In *Proceedings of Computational Linguistics in the North East*.
- Goodall, G. 2009. *Parallel Structures in Syntax: Coordination, Causatives and Restructuring*. Cambridge University Press.
- Graff, D. and C. Cieri. 2003. English Gigaword. Linguistic Data Consortium, Philadelphia.
- Grosz, B., A. K. Joshi, and S. Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics* 21(2):203–225.
- Hardt, D. 1997. An empirical approach to VP ellipsis. *Computational Linguistics* 23(4):525–541.
- Hirshman, L. and N. Chinchor. 1997. MUC-7 Coreference Task Definition. Version 3.0. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*. Applications International Corporation.

- Hobbs, J. R. and A. Kehler. 1998. A theory of parallelism and the case of VP ellipsis. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL-98)*, pages 394–401.
- Knight, K. and D. Marcu. 2002. Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artificial Intelligence* 139(1):91–107.
- Krippendorff, K. 2010. Krippendorff’s alpha. In N. Salkind, ed., *Encyclopedia of Research Design*, pages 669–674. Thousand Oaks, CA: SAGE Publications, Inc.
- Lee, H., A. Chang, Y. Peirsman, N. Chambers, M. Surdeanu, and D. Jurafsky. 2013. Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics* 39(4):885–916.
- Manning, C. D., M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.
- McShane, M. 2000. Hierarchies of parallelism in elliptical Polish structures. *Journal of Slavic Linguistics* 8:83–117.
- McShane, M. and S. Nirenburg. 2012. A knowledge representation language for natural language processing, simulation and reasoning. *International Journal of Semantic Computing* 6:3–23.
- Mitkov, R. 2001. Outstanding issues in anaphora resolution. In *Computational Linguistics and Intelligent Text Processing*, pages 110–125. Springer.
- Nielsen, L. A. 2005. *A corpus-based study of verb phrase ellipsis identification and resolution*. Ph.D. thesis, King’s College London.
- Nirenburg, S. and V. Raskin. 2004. *Ontological Semantics*. Cambridge, Mass.: The MIT Press.
- Olsson, F. 2004. A survey of machine learning for reference resolution in textual discourse. Tech. Rep. T2004:02. ISSN 1100-3154. ISRN:SICS-T-2004/02-SE., SICS.
- Poesio, M., R. Stevenson, B. di Eugenio, and J. Hitzeman. 2004. Centering: A parametric theory and its instantiations. *Computational Linguistics* 30:309–363.
- Stoyanov, V., N. Gilbert, C. Cardie, and E. Riloff. 2009. Conundrums in noun phrase coreference resolution: Making sense of the state-of-the-art. In *Proceedings of Joint Conference of the 47th Annual Meeting of the Association of Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*.
- Vanderwende, L., H. Suzuki, C. Brockett, and A. Nenkova. 2007. Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing and Management* 43(6):1606–1618.

- Webber, B. L. 1988. Discourse deixis: reference to discourse segments. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics (ACL '88)*, pages 113–122.