Annotating Causal Language Using Corpus Lexicography of Constructions

Jesse Dunietz
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213, USA
jdunietz@cs.cmu.edu

Lori Levin and **Jaime Carbonell**
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15213, USA
{lsl,jgc}@cs.cmu.edu

Abstract

Detecting and analyzing causal language is essential to extracting semantic relationships. To that end, we present an annotation scheme for English causal language (not metaphysical causality), and discuss two methodologies for annotation. The first uses only a coding manual to train annotators in distinguishing causal from non-causal language. To address low inter-coder agreement, we adopted a second methodology, in which we first created a causal language *constructicon* based on corpus analysis, then required annotators only to annotate instances based on the constructicon. (This resembles the methodology used for annotating the FrameNet and PropBank corpora.) Our contributions, in addition to the annotation scheme itself, are methodological: we discuss when constructicon-based methodology is appropriate, and address the validity of annotation schemes that require expert-level metalinguistic awareness.

1 Introduction

Information extraction relies on identifying and analyzing the semantic relationships expressed in text. One of the most important kinds of relationship to extract is causality: we think about the world around us in terms of causation, and we often consult texts about what causes, enables, or prevents some phenomenon (e.g., medical symptoms, political events, or interpersonal actions). Unsurprisingly, causal language is also ubiquitous; Conrath et al. (2014) found that in French, causation constituted 33% of the relations expressed between verbs.

Despite its centrality to our thought and language, causal relationships are not captured well by standard semantic representations. The linguistic expression of causal relations varies greatly (Wolff et al., 2005), ranging from verbal propositions to discourse relations to arbitrarily complex constructions. There is no one standard representation scheme that can handle all of these types of semantics, making it difficult to analyze and extract causal relationships in a coherent, comprehensive manner.

Filling this gap requires grappling with some of the most difficult issues in language annotation. Causation is a complex concept, heavily discussed in philosophical and psychological circles. Its boundaries are fuzzy: causation is a psychological construct that we use to explain the world around us, and it does not perfectly match either empirical reality or the language we use to describe it (see Neeleman and Van de Koot [2012]). Furthermore, causation is intertwined with many other dimensions of meaning, such as temporal relations, counterfactuals, factivity, and negation. This raises important questions about how to carve out a semantic space for an annotation scheme to meaningfully represent. It also raises practical questions about how to guide annotators to sensible decisions in such a domain.

In this paper, we describe three primary contributions toward coping with the complexity of annotating causal language. First, drawing on principles from construction grammar, we present a new annotation scheme for causal language. The scheme provides a uniform representation for a wide spectrum of causal language, while still allowing for semantically relevant dimensions of variation. It attempts to

Proceedings of LAW IX - The 9th Linguistic Annotation Workshop, pages 188–196,
Denver, Colorado, June 5, 2015. ©2015 Association for Computational Linguistics

limit the complexity of annotation by focusing not on the hairy metaphysics of causation, but on the assertions about causation that are explicit in the language. We ultimately plan to use this scheme in an automated causal information extraction system.

Our second contribution is to compare two approaches to annotating causality, one using an annotation manual only and the other using a *constructicon* developed by an expert along with an annotation manual. The constructicon-based methodology is similar to the two-stage methodology used in PropBank (Palmer et al., 2005) and FrameNet (Baker et al., 1998) annotations: an initial phase of corpus lexicography produces a lexicon, followed by a second phase in which annotators identify instances of the lexical frames in a corpus. In our case, the “lexicon” is a list of English constructions that conventionally express causality. We also offer suggestions for when such an approach may be appropriate.

Finally, we discuss the broader implications of our experience for difficult annotation tasks. In particular, we address the concern of arbitrariness in schemes which can only be successfully be applied by experts or highly trained annotators.

2 Related Work

2.1 Annotating Real-World Causal Relations

Several previous projects have attempted to annotate causation in text. Many of these have focused on annotating the causal relations that exist in the real world, rather than causal language.

SemEval 2007 included a task (Girju et al., 2007) concerning classifying semantic relations between nominals, including causal relations. As part of this task, the organizers provided a dataset tagged with noun-noun relations. However, this task relied on a less precise, common-sense notion of real-world causation, and the annotations do not indicate the causal connectives, presumably because real-world causal relationships may not be indicated in the text. The SemEval data also limited the causes and effects to nouns (in our experience, they are often clauses).

Grivaz (2010) finds that human annotators struggle to apply standard philosophical tests to make binary decisions about the presence of causation in a text segment. She suggests alternative criteria, which we take into account in our coding manual. Many of her criteria, however, are concerned with how people identify real-world causal relationships, rather than how speakers or writers explicitly invoke the concept of causality.

The Richer Events Description schema has also incorporated cause/effect relations (Ikuta et al., 2014). This effort, too, is concerned with bringing annotators to agreement on what counts as real-world causation. It is also limited to event-event relations, even though causal language often describes states or objects as causes or effects.

2.2 Annotating Causal Language

Other projects have, to a greater or lesser extent, focused on annotating *stated* causal relationships, much as we have. In general, our scheme attempts to be more precise in its definitions, more general in its scope, and more rich in its representational capacity than these prior works.

The Penn Discourse TreeBank (PDTB; Prasad et al., 2008) includes several relation types that are relevant to causation (primarily CAUSE and REASON). Its representation of causal relations is limited in three important ways that we attempt to overcome. First, it does not capture the subtleties of different types of causal relationships. Second, it is limited to discourse relations, and so excludes other realizations of the relationship (e.g., verb arguments). Finally, its relation hierarchy fails to capture overlaps between the semantics of different discourse phenomena (e.g., hypotheticals may also be causal).

Closer to our work is the scheme proposed by Mirza et al. (2014), who base their representation on Talmy’s “force dynamics” model of causation (Talmy, 1988). Their model is rich enough to capture linguistic triggers of causation, as well as causes and effects. It particularly follows Wolff’s (2005) taxonomy of expressions of causation. However, like the PDTB, it does not distinguish the different types of causal relationships. It also does not rigorously define what it counts as causal, and like Ikuta’s work, it is limited to event-event relations.

The project most similar in spirit to ours is BioCause (Mihăilă et al., 2013), which provides an annotation framework for causal relations in biomedical texts. The BioCause framework, like ours, marks the connective and argument spans and the direction of causality. The primary difference between Bio-

Cause and our project is that ours aims to be more general in scope. As such, our scheme also does not examine some kinds of domain-specific language that BioCause includes (e.g., upregulation). In a sense, our project may be thought of as a generalization of BioCause to broader domains, and also an attempt to pin down more precisely what kinds of relationships to annotate as causal.

3 Causal Language Annotation Scheme

3.1 Annotation Scheme Design Philosophy

For the purposes of this project, we are interested in studying specifically *causal language* – the language used to appeal to psychological notions of causality. We are **not** concerned with identifying relationships that are causal in some "true" metaphysical sense; what characterizes true causation is a highly contentious topic within philosophy (Schaffer, 2014; Dowe, 2008). We believe that focusing on this question has unnecessarily confounded previous attempts at annotating causation in text.

Instead, we are concerned only with *what the text asserts* – causal *language* and what is meant by it. If and only if the text explicitly appeals to some psychological notion of promoting or hindering, then the relationship it asserts is one we want to represent, whether or not it is metaphysically accurate.

Consider, for example, the sentence "She must have met him before, because she recognized him yesterday." Few philosophers would say it expresses a "truly" causal relationship, but it does appeal to the psychological notion of causation. (The category of INFERENCE, described below, was included specifically to handle cases like this.)

Although the boundaries of causality are not well-defined, we wished to study causal language in isolation to the extent we could. We therefore designed the annotation scheme to exclude language that incorporates other elements of meaning beyond causality, as well as language whose causal interpretation is ambiguous or merely suggestive. However, we also designed it to be composable with other components of semantic analysis: negation, aspect, hedging, and so on. We assume that other annotation schemes will represent these aspects, and that this additional information may alter the semantics of the causal relationship as a whole.

Our current focus is English only. We believe that the basic components of the annotation scheme should apply in other languages, but many adjustments to the criteria for inclusion would be needed.

3.2 Defining Causal Language

We use the term *causal language* to refer to clauses or phrases in which one event, state, action, or entity (the cause) is *explicitly presented as* promoting or hindering another (the effect). The cause and effect must be deliberately related by an explicit causal connective. (As emphasized above, the words "presented as" are essential to this definition.)

Causal relations can be expressed in English in many different ways. In this project, we exclude:

- **Causal relationships with no lexical trigger.** We do not annotate implicit causal relationships ("zero" discourse connectives). We expect our work to be compatible with other work on such relationships, such as the implicit relations in the PDTB and systems for recovering those relationships (Conrath et al., 2014).

- **Connectives that lexicalize the means or the result of the causation.** For example, *kill* can be interpreted as "cause to die," but it encodes the result, so we exclude it. This decision was made to allow the scheme to focus specifically on language that expresses causation. If lexical causatives were included, nearly every transitive verb in the English language would have to be considered causal; it would be impossible to disentangle causation as a semantic phenomenon with its own linguistic realizations. It would also be impossible to annotate the cause and effect separately from the connective.[1]

  Omitting lexical causatives is consistent with previous causal language annotation schemes (e.g., Mirza et al. [2014]), though we are not aware of previous attempts to define what must be lexicalized for a verb to be excluded.

- **Connectives that assert an unspecified causal relationship.** "Smoking is linked to cancer"

[1] If lexical connectives are ever desired, the PropBank or FrameNet lexicon could be augmented to indicate which verb senses are causal, and the associated corpus could then act as a supplemental causal language corpus.

does not specify what sort of causal link is present, so we do not annotate it.

- **Temporal language** (e.g., "After I drank some water, I felt much better"). These instances are often extremely ambiguous ("after" can be purely temporal). Even when they are unambiguously causal, the causal relationship is clear not from causal *language*, but from real-world knowledge about the events presented.

3.3 Anatomy of a Causal Language Instance

For each instance of causal language that meets these criteria, we annotate three spans (any of which may be non-contiguous):

- **The causal connective** – the lexical items in the construction signalling the causal relationship. Following the basic ideas of construction grammar (Fillmore et al., 1988), the connective may be any surface linguistic pattern conventionally used to indicate causation. Such constructions generally have at least two open slots (for cause and effect). The connective annotation includes all words whose lemmas appear in every instance of the construction.
- **The cause.** Causes are generally events or states of affairs, expressed as complete clauses or phrases. Sometimes, however, an actor, but not an action, is presented as the cause (e.g., "I prevented a fire."). In such cases, we take the actor to be metonymic for the action, and accordingly annotate the actor as the cause.
- **The effect.** Also generally an event or state of affairs, expressed as a complete clause/phrase.

In general, the spans of the arguments do not overlap with the spans of the connectives (though there are some exceptions).

3.4 Types of Causation

We distinguish four different types of causal relationships, each of which can have subtly different semantics. Examples of each are given in Table 1.

Causation type	Example
CONSEQUENCE	*We are in serious economic trouble* because of **inadequate regulation**.
MOTIVATION	**We don't have much time**, so *let's move quickly*.
PURPOSE	To **strengthen our company**, *we must set clearer policies*.
INFERENCE	*This car was driven recently*, because **the hood is still hot**.

Table 1: Examples of each of the four types of causal language (with **causes** in bold and *effects* in italics).

CONSEQUENCE instances assert that the cause naturally leads to the effect via some chain of events, without highlighting the conscious intervention of any agent. The majority of instances are CONSEQUENCEs (see Table 2).

MOTIVATION instances assert that some agent perceives the cause, and therefore consciously thinks, feels, or chooses something. Again, what is important for this scheme is how the relationship is presented, so an instance is MOTIVATION only if it frames the relationship in a way that highlights an agent's decision or thought.

PURPOSE instances assert that an agent chooses the effect out of a desire to make the cause true. What distinguishes PURPOSEs from MOTIVATIONs is whether the motivating argument is a fact about the world or an outcome the agent hopes to achieve.

Note that there is a confusing duality in PURPOSEs. The desire for a particular outcome (e.g., "to strengthen our company") motivates (causes) the effect ("we must set clearer policies"). But from another perspective, having clearer policies is a cause whose effect may be strengthening the company. We choose to focus on the first of these relationships because we take this to be the primary relationship expressed by language such as "in order to."

INFERENCE instances borrow the language of CONSEQUENCE, but they do not assert an actual chain of events from cause to effect. Instead, they present the cause as evidence or justification for the effect (*epistemic causation*).

3.5 Degrees of Causation

In principle, causal relationships lie on a spectrum from total prevention to total entailment. Wolff et al. (2005) discretize this spectrum into three categories: CAUSE, ENABLE, and PREVENT. In practice, however, we found that annotators were able to reliably

	In subcorpus annotated with:	
	Manual only	Constructicon
CONSEQUENCE	66	33
MOTIVATION	18	11
PURPOSE	4	21
INFERENCE	0	4
Total	88	69

Table 2: Number of instances of each causation type in the subcorpora used for IAA. (Counts are from the first author's annotations.)

distinguish only positive and negative causation. We therefore annotate the degree of each instance as either FACILITATE or INHIBIT. (We hope to return to finer-grained distinctions of degree in future work.)

3.6 Tools and Data

We performed all annotations using BRAT (Stenetorp et al., 2012), a web-based annotation tool. A sample annotation is shown in Figure 1.

For our corpus, we randomly selected documents from the Washington section of the New York Times corpus (Sandhaus, 2008) from the year 2007. We found that the political nature of these documents lent itself to more frequent use of causal language. At present, we have annotated ~1200 sentences in total, containing ~400 instances of causal language.

4 Initial Annotation Process: Coding Manual Only

In the design phase of our project, we developed a coding manual for this annotation scheme, working with three annotators to identify and decide on difficult cases. Once we felt the manual was ready for large-scale annotation, we spent several weeks training a previously uninvolved annotation expert to apply the scheme. The first author's annotations on 201 sentences (containing about 88 instances of causal language) were then compared against the new annotator's to determine inter-annotator agreement. The counts of different causation types are shown in Table 2.

Under this process, annotators were expected to consider all principles and special cases laid out in the manual for each decision: whether something counted as causal language at all, what words should be included in the connective, and what the argument spans should be. Decision trees were provided to determine the degree and type of the instance.

	Partial overlap:	
	Allowed	Excluded
Connectives (F_1)	0.70	0.66
Degrees (κ)	0.87	0.87
Causation types (κ)	0.25	0.29
Argument spans (F_1)	0.94	0.83
Argument labels (κ)	0.92	0.94

Table 3: Inter-annotator agreement for the coding-manual-only approach, showing the middling degree of reliability achieved for connectives and causation types.

The difference between the two columns is that for the left column, we counted two annotation spans as a match if at least a quarter of the larger one overlapped with the smaller; for the right column, we required an exact match.

κ scores indicate Cohen's kappa. Each κ score was calculated only for spans that agreed (e.g., degrees were only compared for matching connective spans).

5 Initial Annotation Results and Difficulties

Our initial results (Table 3) did not seem to reflect our many iterations of feedback with the new annotator. For connectives that matched, the argument annotations agreed fairly well, as did the degrees. But the agreement rate for the connectives themselves was only moderately good, and agreement on causation types was abysmal.

Furthermore, the annotator, who had more than 30 years of annotation experience in other tasks, reported that she had found the process torturous and time-consuming, and that she still did not feel confident in her choices. Even to achieve the results in Table 3, the annotator had to ask several clarification questions about specific constructions. This matched the experience of the earlier annotators who had helped us develop the scheme: they felt the guidelines made sense, and for any given annotation they could reach consensus via discussion, but even after working with the scheme for months, annotating still felt difficult and uncertain.

These results raised two important questions. The first was a matter of procedure: what could we do

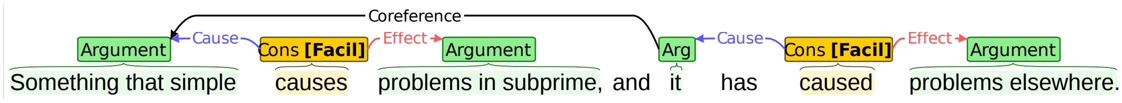
Coreference
Argument
Cause
Cons [Facil]
Effect
Argument
Arg
Cause
Cons [Facil]
Effect
Argument
Something that simple causes problems in subprime, and it has caused problems elsewhere.

Figure 1: Two fully-annotated instances of causal language in BRAT. Coreference links are included in the annotation only for arguments that consist entirely of a pronoun.

to improve the annotation process and reliability? The second question was more fundamental: even assuming we could improve the agreement scores, how should we interpret the fact that annotators were struggling so? If the scheme was still unintuitive after so much training, was it even meaningful at all?

In the next two sections, we address each of these questions in turn.

6 Modularizing the Annotation Process with Corpus Lexicography

The biggest factor dragging down annotators' comfort seemed to be the sheer number of decisions they had to make. In particular, we were expecting them to mentally redraw for every possible connective the fuzzy line between causal and non-causal, keeping in mind the entire gestalt of guidelines and special cases. It is no surprise that this task felt overwhelming, especially given that even once they had decided an instance was causal, they still faced decisions about annotation spans, causation type, and degree.

Much of this effort is in fact redundant. Most connectives in a text will be familiar, and the uses of any given connective are fairly consistent. Accordingly, once a decision about a linguistic pattern has been made once, that decision can often be applied to future instances of the pattern.

Accordingly, we split the annotation process into two phases. In the first phase, we compiled a *constructicon* – a simple list of known causal constructions – by manually cataloguing all connectives seen so far (including in the original annotation set). This catalog could then be quickly consulted whenever annotators encountered a potential connective. As exemplified in Table 4, the catalog gives the word senses for which each connective pattern applies, as well as possible variants, which words to include in the connective span, the degree the connective indicates, and in some cases restrictions on its causation type. Building the constructicon thus requires the same difficult decisions, but these decisions can be made once in consultation with others, and then applied repeatedly to new instances of each pattern.

Connective pattern	⟨cause⟩ prevents ⟨effect⟩ from ⟨effect⟩
WordNet senses	prevent.verb.01 prevent.verb.02
Annotatable words	prevent, from
Degree	INHIBIT
Type restrictions	Not PURPOSE

Table 4: A sample entry in the constructicon.

The constructicon currently includes 166 constructions, covering 79 lexically distinct connectives (e.g., "prevent ___" and "prevent ___ from ___" are the same connective but distinct constructions).

In the second phase, annotators used the constructicon to label novel text. The task primarily now consisted of recognizing known patterns and making sure that the word senses used in the text matched the senses for which the patterns were defined.

Of course, there is a cycle in this process: if annotators spot a plausible connective that is not in the constructicon, they can propose it to be added. But given the relative rarity of novel connectives, this is not the annotators' primary task.

We expect to release both the constructicon and an expanded corpus based on it at a later date.

6.1 Lexicography-Based Annotation Results

Using this method, we trained another annotator for about a day. After just two rounds of annotation with feedback, the first author and the new annotator both used the constructicon to annotate a new dataset of 260 sentences, drawn from the same corpus, containing 69 instances of causal language.[2]

We expected inter-annotator agreement to decrease compared to our previous attempt. The new

[2] We did not reuse the same dataset because the first author had become too familiar with it and it had informed the constructicon, so it would not have been a meaningful test.

	Partial overlap:	
	Allowed	**Excluded**
Connectives (F_1)	0.78	0.70
Degrees (κ)	1.0	1.0
Causation types (κ)	0.82	0.80
Argument spans (F_1)	0.96	0.86
Argument labels (κ)	0.98	0.97

Table 5: Inter-annotator agreement results with annotators using the constructicon. See Table 3 for a fuller description of how these statistics were computed.

annotator had far less annotation experience, and he had received a fraction of the training on this task. Additionally, we had fewer coded instances, which tends to cause κ scores to drop, and it seemed likely that the lower density of causal language would make it harder to spot the occasional instance.

In fact, our results (shown in Table 5) improved on our initial results in several important respects. First, there was a modest increase in F_1 for connectives. Second, agreement on causation types was now excellent. Third, all other metrics, even those that had already been high, improved slightly. And perhaps most significantly, these results were achieved with a fraction of the training time – a day instead of weeks – and the annotator found annotating quite painless.

Given that these results were computed on a different dataset, it is possible that the improvements are not as great as they seem. Nonetheless, the difference in annotator comfort was striking, and we believe that both datasets are representative.

Of course, the lexicography work itself still takes significant effort – effort that we were able to shortcut somewhat by mining our existing annotations to build the constructicon. But in general, the lexicography could be done in parallel with refining the scheme itself, as trial datasets are annotated.

6.2 When is Lexicography Appropriate?

The lexicography-based approach to semantic annotation is not new, of course. Several high-profile annotation projects have used it successfully, most notably PropBank and FrameNet. But it is a relatively uncommon approach for projects to take. Our experience suggests that although lexicography may not work well for every annotation effort, it may be more widely useful than current practice would indicate.

The essential question, then, is what characteristics make a project a good fit for corpus lexicography. Our experience here is limited, but one feature of our project seems to have made it particularly amenable to this approach: without a constructicon, annotators had to make the same decisions repeatedly. This was the core reason why the constructicon was useful; a constructicon would not save any work if it did not codify frequently made decisions.

Of course, a lexicography-based approach intensifies concerns about meaningfulness. Adopting a lexicon may increase inter-annotator agreement, but what annotators are agreeing on is more constrained. A generous reading is that that the experts who compiled the lexicon have helped less-expert annotators make more accurate choices. But there is a less charitable reading, as well: if such constraints are needed for agreement, perhaps the annotation scheme fails to capture meaningful semantic categories – perhaps it is merely a fiction of the minds of its designers. It is to this concern that we turn next.

7 What Does Low Non-Expert Agreement Say About Validity?

What imparts validity to an annotation scheme is a fundamental question that haunts every annotation project. Even a well-thought-out scheme can include arbitrary, empirically meaningless decisions, which would seem to undermine the scheme's value as a description of a real linguistic phenomenon.[3]

This risk of arbitrariness is precisely what appears to bother Riezler (2014) in his discussion of circularity in computational linguistics: it is entirely possible that an annotation scheme has high inter-annotator agreement and can even be reproduced by software, and yet the scheme is empirically empty. The agreement can be achieved simply by developing a shared body of implicit, arbitrary theoretical assumptions among expert or intensively trained coders. Meanwhile, the fact that the annotations can be reproduced automatically shows only that the theory can be expressed both as an annotation scheme and as an annotation machine, not that it encapsulates something meaningful.

[3] Similar questions arise in designing and assessing tests for social science research (Trochim, 2006).

Thus, the problem of arbitrary assumptions raises especially serious questions about any scheme for which expertise seems to be required. If a scheme requires expert input or intensive training to reach agreement, that seems to suggest that the scheme is really a "stone soup" of theoretical, possibly arbitrary assumptions among the experts.

One tempting solution is Riezler's first suggestion for breaking circularity: using naive coders, such as crowdsourced annotators. The instructions that convey the scheme to the coders, who do not share the same theoretical assumptions, constitute a second theory that the original theory can be grounded in. This, Riezler implies, would demonstrate the empirical reality of the theory behind the scheme, which he presumably would argue confirms its value.

For many schemes, high agreement among naive coders may indeed break the circularity of the scheme. But as our lexicography-based approach highlights, this solution may not address the deeper problem of arbitrariness. Consider an annotation guide that relies on a lexicon to save the coders decisions. It is debatable whether this would qualify as a sufficiently different description of the theory to break circularity. Either way, though, if the original scheme was arbitrary, the arbitrariness still remains, even if naive coders achieve high agreement. The arbitrary rules are no longer hidden in the heads of the annotators, but instead they are baked directly into the annotation guidelines as pre-made decisions. It seems, then, that the possibility of crowdsourcing (or, more generally, non-expert annotation) is not *sufficient* to make a scheme worthwhile.

Some (though notably not Riezler) have argued that disagreement among naive coders demonstrates the empirical emptiness of a scheme – i.e., that the possibility of crowdsourcing is still a *necessary* condition for a scheme's validity. (The concerns we raised above suggest this argument, as well.) This argument is also problematic, because it assumes that naive coders' explicit knowledge accurately reflects how their language works. That may seem reasonable – after all, naive coders are competent users of the language. But in practice, there is no reason to expect the average person to have meta-linguistic awareness, any more than one would expect a baseball player – a competent user of physics – to correctly identify the physics phenomena at work when he swings. The fact that expertise is required to precisely describe a phenomenon does not mean that the phenomenon is not empirically real.

If, consequently, agreement among naive coders is neither necessary nor sufficient to ascribe value to an annotation scheme, how do we proceed?

One way out is Riezler's second proposal: extrinsic task-based evaluation. If an annotation scheme is useful for a particular downstream NLP task – e.g., information extraction – then in some sense it is irrelevent whether the scheme is arbitrary; it at least correlates with the truth enough to be practically useful. We hope our scheme for causal language will fall into this category by proving useful, both directly to humans seeking causal information and for downstream information extraction.

Another way out is a type of usefulness that Riezler does not discuss. Often, simply attempting to formalize a phenomenon yields insights into some aspect of language, even if the formalization is empirically questionable.

It could be, for example, that our causal language scheme invents empirically meaningless semantic categories. However, it may still suggest hypotheses about how people use certain causal constructions. For instance, how often people talk about inhibiting vs. facilitating may vary dramatically depending on the genre. If validated, such an observation would yield valuable insights about language use and perhaps psychology – insights we would not have even thought to look for without the annotation scheme.

In short, then, we do not believe that low agreement among naive coders (or a need for expert guidance in decision-making, such as a lexicon) necessarily impugns the value of an annotation scheme as a whole. Accordingly, we hope that our suggestion of construction-based lexicography will help others build annotation schemes and corpora that are valuable by the criteria we have outlined. In our own future work, we hope to demonstrate that our causal language scheme meets these criteria, as well.

Acknowledgments

We would like to thank Nora Kazour, Mike Mordawanec, Spencer Onuffer, Donna Gates, Jeremy Doornbos, and Chu-Cheng Lin for all their help with annotations and annotation scheme refinement.

References

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The Berkeley FrameNet Project. In *Proceedings of the 17th International Conference on Computational Linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

Juliette Conrath, Stergos Afantenos, Nicholas Asher, and Philippe Muller. 2014. Unsupervised extraction of semantic relations using discourse cues. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2184–2194. Dublin City University and Association for Computational Linguistics.

Phil Dowe. 2008. Causal processes. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Fall 2008 edition. `http://plato.stanford.edu/archives/fall2008/entries/causation-process/`.

Charles J. Fillmore, Paul Kay, and Mary Catherine O'Connor. 1988. Regularity and idiomaticity in grammatical constructions: The case of let alone. *Language*, 64(3):501–538, September.

Roxana Girju, Stan Szpakowicz, Preslav Nakov, Peter Turney, Vivi Nastase, and Deniz Yuret. 2007. SemEval-2007 task 04: Classification of semantic relations between nominals. In *In Fourth International Workshop on Semantic Evaluations (SemEval-2007)*.

Cécile Grivaz. 2010. Human judgements on causation in French texts. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*. European Languages Resources Association (ELRA).

Rei Ikuta, Will Styler, Mariah Hamang, Tim O'Gorman, and Martha Palmer. 2014. Challenges of adding causation to Richer Event Descriptions. In *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 12–20. Association for Computational Linguistics.

Claudiu Mihăilă, Tomoko Ohta, Sampo Pyysalo, and Sophia Ananiadou. 2013. BioCause: Annotating and analysing causality in the biomedical domain. *BMC Bioinformatics*, 14(1):2.

Paramita Mirza, Rachele Sprugnoli, Sara Tonelli, and Manuela Speranza. 2014. Annotating causality in the TempEval-3 corpus. In *Proceedings of the EACL 2014 Workshop on Computational Approaches to Causality in Language (CAtoCL)*, pages 10–19.

Ad Neeleman and Hans Van de Koot, 2012. *The Theta System: Argument Structure at the Interface*, chapter The Linguistic Expression of Causation, pages 20–51. Oxford University Press.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse TreeBank 2.0. In Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, and Daniel Tapias, editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May. European Language Resources Association (ELRA).

Stefan Riezler. 2014. On the problem of theoretical terms in empirical computational linguistics. *Computational Linguistics*, 40(1):235–245.

Evan Sandhaus. 2008. The New York Times annotated corpus. *Linguistic Data Consortium, Philadelphia*.

Jonathan Schaffer. 2014. The metaphysics of causation. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Summer 2014 edition. `http://plato.stanford.edu/archives/sum2014/entries/causation-metaphysics/`.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. BRAT: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107. Association for Computational Linguistics.

Leonard Talmy. 1988. Force dynamics in language and cognition. *Cognitive Science*, 12(1):49–100.

William M. Trochim. 2006. Reliability & validity. In *The Research Methods Knowledge Base, 2nd Edition*. `http://www.socialresearchmethods.net/kb/relandval.php`.

Phillip Wolff, Bianca Klettke, Tatyana Ventura, and Grace Song. 2005. Expressing causation in English and other languages. In Woo-kyoung Ahn, Robert L. Goldstone, Bradley C. Love, Arthur B. Markman, and Phillip Wolff, editors, *Categorization inside and outside the laboratory: Essays in honor of Douglas L. Medin*, pages 29–48. American Psychological Association, Washington, DC, US.