

Using Hedge Detection to Improve Committed Belief Tagging

Morgan Ulinski and Seth Benjamin and Julia Hirschberg

Department of Computer Science

Columbia University

New York, NY, USA

{mulinski@cs., sjb2190@, julia@cs.}columbia.edu

Abstract

We describe a novel method for identifying hedge terms using a set of manually constructed rules. We present experiments adding hedge features to a committed belief system to improve classification. We compare performance of this system (a) without hedging features, (b) with dictionary-based features, and (c) with rule-based features. We find that using hedge features improves performance of the committed belief system, particularly in identifying instances of non-committed belief and reported belief.

1 Introduction

Hedging refers to the use of words, sounds, or constructions that add ambiguity or uncertainty to spoken or written language. Hedges are often used by speakers to indicate lack of commitment to what they say; so, the ability to classify words and phrases as hedges is very relevant to the task of committed belief tagging—that is, determining the level of commitment a speaker has toward the belief expressed in a given proposition. A major challenge in identifying hedges is that many hedge words and phrases are ambiguous. For example, In (1), *around* is used as a hedge, but not in (2).

- (1) She weighs **around** a hundred pounds.
- (2) Suddenly she turned **around**.

Currently there are few corpora annotated for hedging, and these are in a limited number of genres. In particular, there is currently no corpus of *informal* language annotated with hedge behavior. Acquiring expert annotations on text in other genres can be time consuming and may be cost prohibitive, which is an impediment to exploring how hedging can help with applications based on text and other genres. In this paper, the application we focus on is committed belief tagging on a corpus

of forum posts. Since we currently lack a labeled hedging corpus in this genre, we introduce a new method for disambiguating potential hedges using a set of manually-constructed rules. We then show that detecting hedges using this method improves the performance of a committed belief tagger.

In Section 2, we discuss related work. In Section 3, we describe how we identify hedges. We describe the committed belief tagger used for our experiments in Section 4. In Section 5, we describe our experiments and our results. We conclude and discuss future work in Section 6.

2 Related Work

Most work on hedge detection has focused on using machine learning models based on annotated data, primarily from the domain of academic writing. The CoNLL-2010 shared task on learning to detect hedges (Farkas et al., 2010) used the BioScope corpus (Vincze et al., 2008) of biomedical abstracts and articles and a Wikipedia corpus annotated for “weasel words.” Most CoNLL-2010 systems approach the task as a sequence labeling problem on the token level (e.g. Tang et al. (2010)); others approached it as a token-by-token classification problem (e.g. Vlachos and Craven (2010)) or as a sentence classification problem (e.g. Clausen (2010)).

Our approach is closest to Velldal (2011), a follow-up to CoNLL-2010 which frames the task of identifying hedges as a disambiguation problem in which all potential hedge cues are located and then subsequently disambiguated according to whether they are used as a hedge or not. However, our work differs in that we use a set of manually-constructed rules to disambiguate potential hedges rather than a machine learning classifier. Using a rule-based rather than machine-learning approach allows us to apply our hedge detection method to

Relational Hedges	Propositional Hedges
according to, appear , arguably, assume , believe , consider , could, doubt , estimate, expect, feel , find , guess, hear, I mean, I would say, imagine , impression , in my mind, in my opinion, in my understanding, in my view, know , likely , look like, looks like, may , maybe, might , my thinking, my understanding, necessarily, perhaps, possibly, presumably, probably, read, say, seem, seemingly, should , sound like, sounds like, speculate, suggest, suppose , sure , tend , think, understand, unlikely , unsure	a bit, a bunch, a couple, a few, a little, a whole bunch, about , allegedly, among others, and all that, and so forth, and so on, and suchlike, apparently, approximately, around, at least, basic, basically, completely , et cetera, etc, fair , fairly , for the most part, frequently, general , generally, in a way, in part, in some ways, kind of, kinda, largely, like , mainly, more or less, most , mostly, much, occasionally, often , partial , partially, partly, possible , practically, pretty , pretty much, probable, rarely, rather , really , relatively, rough, roughly , seldom, several, something or other, sort of, to a certain extent, to some extent, totally , usually, virtually

Table 1: List of (potential) hedge words and phrases.

a corpus of forum posts that has not been annotated with hedge information. Our work also differs from previous efforts in that we are interested not just in the problem of hedge detection itself, but in its application to committed belief tagging.

3 Identifying Hedge Terms

We first compiled a dictionary of 117 potential hedge words and phrases. We began with the hedge terms identified during the CoNLL-2010 shared task (Farkas et al., 2010), along with synonyms of these terms extracted from WordNet. This list was further expanded and edited through consultation with the Linguistic Data Consortium (LDC) and other linguists. For each hedge term in our dictionary, we wrote definitions defining the hedging and non-hedging usages of the term. We use these definitions as the basis for the rules in our hedge classifier.

This hedging dictionary is divided into *relational* and *propositional* hedges. As described in Prokofieva and Hirschberg (2014), relational hedges have to do with the speaker’s relation to the propositional content, while propositional hedges are those that introduce uncertainty into the propositional content itself. Consider the following:

- (3) I **think** the ball is blue.
- (4) The ball is **sort of** blue.

In (3), *think* is a relational hedge. In (4), *sort of* is a propositional hedge.

Our baseline hedge detector is a simple, *dictionary-based* one. Using our dictionary of potential hedge terms, we look up the lemma of each token in the dictionary and mark it as a hedge if

found. This procedure, however, does not take into account the inherent ambiguity of many of the hedge terms. To handle this ambiguity, we implemented *rule-based* hedge detection. The rule-based system disambiguates hedge vs. non-hedge usages using rules based on context, part-of-speech, and dependency information.

The full list of hedge words and phrases in our dictionary is shown in Table 1. The hedge terms for which we have written rules are shown in bold; the rule-based system classifies others as hedges by default. Table 2 shows a sample of the rules, with examples of hedging and non-hedging uses.

We evaluate both dictionary-based and rule-based approaches in a committed belief tagger.

4 Committed Belief Tagger

We employ the committed belief tagger described in Prabhakaran et al. (2010) and as Sytem C in Prabhakaran et al. (2015). This tagger uses a quadratic kernel SVM to train a model using lexical and syntactic features. Tags are assigned at the word level; the tagger identifies tokens denoting the heads of propositions and classifies each proposition as one of four belief types:

- **Committed belief (CB):** the speaker-writer believes the proposition with certainty, e.g.
 - (5) The sun will rise tomorrow.
 - (6) I know John and Katie went to Paris last year.
- **Non-committed belief (NCB):** the speaker-writer believes the proposition to be possibly, but not necessarily, true, e.g.

Hedge term	Rule	Examples
about	If token t has part-of-speech IN, t is non-hedge. Otherwise, hedge.	Hedge: There are about 10 million packages in transit right now. Non-hedge: We need to talk about Mark.
likely	If token t has relation $amod$ with its head h , and h has part-of-speech N^* , t is non-hedge. Otherwise, hedge.	Hedge: We will likely stay home this evening. Non-hedge: He is a fine, likely young man.
rather	If token t is followed by token 'than', t is non-hedge. Otherwise, hedge.	Hedge: She's been behaving rather strangely. Non-hedge: She seemed in-different rather than angry.
assume	If token t has $ccomp$ dependent, t is hedge. Otherwise, non-hedge.	Hedge: I assume his train was late. Non-hedge: When will the president assume office?
tend	If token t has $xcomp$ dependent, t is hedge. Otherwise, non-hedge.	Hedge: Written language tends to be formal. Non-hedge: Viola tended plants on the roof.
appear	If token t has $xcomp$ or $ccomp$ dependent, t is hedge. Otherwise, non-hedge.	Hedge: The problem appears to be a bug in the software. Non-hedge: A man suddenly appeared in the doorway.
sure	If token t has neg dependent, t is hedge. Otherwise, non-hedge.	Hedge: I'm not sure what the exact numbers are. Non-hedge: He is sure she will turn up tomorrow.
completely	If the head of token t has neg dependent, t is hedge. Otherwise, non-hedge.	Hedge: That isn't completely true. Non-hedge: I am completely sure you will win.
suppose	If token t has $xcomp$ dependent d and d has $mark$ dependent 'to', t is non-hedge. Otherwise, hedge.	Hedge: I suppose the package will arrive next week. Non-hedge: I'm supposed to call if I'm going to be late.
should	If token t has relation aux with its head h and h has dependent 'have', t is non-hedge. Otherwise, hedge.	Hedge: It should be rainy tomorrow. Non-hedge: He should have been more careful.

Table 2: Examples of rules used to disambiguate hedge terms.

(7) It could rain tomorrow.

(8) I think John and Katie went to Paris last year.

- **Reported belief (ROB):** the speaker-writer reports the belief as belonging to someone else, without specifying their own belief or lack of belief in the proposition, e.g.

(9) Channel 6 said it could rain tomorrow.

(10) Sarah said that John and Katie went to Paris last year.

- **Non-belief propositions (NA):** the speaker-writer expresses some cognitive attitude other than belief toward the proposition, such as desire, intention, or obligation, e.g.

(11) Is it going to rain tomorrow?

(12) I hope John and Katie went to Paris last year.

4.1 Hedge Features

For the experiments described in this paper, we add the following additional features to the committed belief tagger:

- **Word features:** based on properties of the current word being tagged. If the word is classified as a hedge by the hedge detector, $HedgeLemma$, and $HedgeType$ are set to the token, lemma, and hedge type (propositional or relational) of the word. Otherwise, these features are null.

- **Dependency features:** based on attributes of words related to the current word by the dependency parse. If the child of a given

word is classified as a hedge by the hedge detector, *HedgeTokenChild*, *HedgeLemmaChild*, and *HedgeTypeChild* are set to the token, lemma, and hedge type (propositional or relational) of the child. Otherwise, these features are null. Likewise, we define *HedgeToken*{Parent,Sibling,DepAncestor}, *HedgeLemma*{Parent,Sibling,DepAncestor}, and *HedgeType*{Parent,Sibling,DepAncestor} if the parent, sibling, or ancestor of the word is classified as a hedge.

- **Sentence features:** based on properties of the sentence containing the current word. If the hedge detector identifies a hedge anywhere in the sentence, *SentenceContainsHedge* is set to true.

5 Experiments and Results

All the experiments reported below use 5-fold cross validation on the 2014 Darpa DEFT Committed Belief Corpus (Release No. LDC2014E55). The documents in this corpus are from English discussion forum data. We compare the performance of the system using (a) no hedge features (b) hedge features obtained using the dictionary-based tagger, and (c) hedge features obtained using the rule-based tagger. Results are shown in Table 3. Note that our baseline results differ slightly from the System C results presented in Prabhakaran et al. (2015) because the training/evaluation datasets used are different. Additionally, our baseline uses no hedge features while System C uses simple word-based hedge features based on an earlier version of our hedging dictionary.

As we might expect, hedge features are most significant in detecting instances of reported belief and non-committed belief. Since these represent only a small portion of the full corpus, the effect on the overall performance is not large. However it is still significant. Using dictionary-based hedge features, we see an increase of 1.82 in the f-measure for ROB as compared to the baseline, from 23.29 to 25.11, and an increase of 2.29 for NCB, from 23.66 to 25.95. The overall f-score increases 0.43, from 67.52 to 69.95. Using rule-based hedge features, the increase compared to the baseline is more significant. For ROB, the f-score shows an increase of 4.14, from 23.29 to 27.43. For NCB, the f-score increases 6.77, from 23.66 to 30.43. The overall increase in the f-score using

the rule-based hedge features is 0.55, from 67.52 to 68.07.

Tag (count)	Precision	Recall	F-measure
ROB (256)	28.02	19.92	23.29
NCB (193)	44.93	16.06	23.66
NA (2762)	77.49	56.34	65.24
CB (4299)	69.80	74.78	72.21
Overall	70.69	64.62	67.52

(a)

Tag (count)	Precision	Recall	F-measure
ROB (256)	30.22	21.48	25.11
NCB (193)	49.28	17.62	25.95
NA (2762)	77.69	56.73	65.58
CB (4299)	70.27	75.04	72.58
Overall	71.18	65.01	67.95

(b)

Tag (count)	Precision	Recall	F-measure
ROB (256)	31.63	24.22	27.43
NCB (193)	50.60	21.76	30.43
NA (2762)	77.89	56.52	65.51
CB (4299)	70.58	74.95	72.70
Overall	71.36	65.07	68.07

(c)

Table 3: Belief results using (a) no hedge detection, (b) dictionary-based hedge detection, and (c) rule-based hedge detection.

6 Summary and Future Work

We have shown that hedge detection can improve the performance of a committed belief tagger, particularly in identifying instances of reported belief and non-committed belief. Using hedge features based on simple dictionary-lookup improves performance compared to the baseline; the addition of manually constructed rules improves performance further. While these results are promising, there are limits to the rule-based approach we have presented. In many cases, it is not straightforward to define a simple rule disambiguating hedge from non-hedge use.

To address these issues, we use Amazon Mechanical Turk to construct a corpus of forum posts labeled with hedge information. Although other labeled corpora exist, these are in other domains and may not apply to the forum data we are using. After finding potential hedges in the forum posts from the 2014 Deft Committed Belief Corpora

3. I'm always **kind of** amused that you guys want to fire all the government workers because they are making too much money, but you have a fit when someone undercuts your salary.

Is the meaning of the word *kind of* closer to:

- type of ("This specimen is a kind of berry as indicated by the seeds located on its skin.")
- to some extent ("It's kind of hard to read them straight up and down like that.")

Figure 1: Example of AMT word disambiguation task.

(Release No. LDC2014E55, LDC2014E106, and LDC2014E125), we present each potential hedge to turkers as a highlighted word or phrase within a sentence. Rather than asking turkers to label the word as a hedge or not, we show the definitions of hedging and non-hedging uses of the term from our hedge dictionary (see Section 3 and ask workers which most closely matches the meaning of the word. Figure 1 shows an example for the phrase *kind of*. In future work, we will use this corpus to evaluate the rule-based hedge detector and to train machine learning classifiers directly from the labeled corpus. By this means, we hope to continue to improve the performance of the committed belief tagger as well.

Acknowledgments

This paper is based upon work supported by the DARPA DEFT program. The views expressed here are those of the author(s) and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

References

- David Clausen. 2010. [Hedgehunter: A system for hedge detection and uncertainty classification](#). In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 120–125, Uppsala, Sweden. Association for Computational Linguistics.
- Richárd Farkas, Veronika Vincze, György Móra, János Csirik, and György Szarvas. 2010. [The conll-2010 shared task: Learning to detect hedges and their scope in natural language text](#). In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 1–12, Uppsala, Sweden. Association for Computational Linguistics.
- Vinodkumar Prabhakaran, Tomas By, Julia Hirschberg, Owen Rambow, Samira Shaikh, Tomek Strzalkowski, Jennifer Tracey, Michael Arrigo, Rupayan Basu, Micah Clark, Adam Dalton, Mona Diab, Louise Guthrie, Anna Prokofieva, Stephanie Strassel, Gregory Werner, Yorick Wilks, and Janyce Wiebe. 2015. [A new dataset and evaluation for belief/factuality](#). In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 82–91, Denver, Colorado. Association for Computational Linguistics.
- Vinodkumar Prabhakaran, Owen Rambow, and Mona Diab. 2010. [Automatic committed belief tagging](#). In *Coling 2010: Posters*, pages 1014–1022, Beijing, China. Coling 2010 Organizing Committee.
- Anna Prokofieva and Julia Hirschberg. 2014. Hedging and speaker commitment. In *5th Intl. Workshop on Emotion, Social Signals, Sentiment & Linked Open Data*, Reykjavik, Iceland.
- Buzhou Tang, Xiaolong Wang, Xuan Wang, Bo Yuan, and Shixi Fan. 2010. [A cascade method for detecting hedges and their scope in natural language text](#). In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 13–17, Uppsala, Sweden. Association for Computational Linguistics.
- Erik Velldal. 2011. [Predicting speculation: a simple disambiguation approach to hedge detection in biomedical literature](#). *Journal of Biomedical Semantics*, 2(5):S7.
- Veronika Vincze, György Szarvas, Richárd Farkas, György Móra, and János Csirik. 2008. [The bioscope corpus: biomedical texts annotated for uncertainty, negation and their scopes](#). *BMC Bioinformatics*, 9(11):S9.
- Andreas Vlachos and Mark Craven. 2010. [Detecting speculative language using syntactic dependencies and logistic regression](#). In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 18–25, Uppsala, Sweden. Association for Computational Linguistics.