# HedgeHunter: A System for Hedge Detection and Uncertainty Classification

**David Clausen**
Department of Linguistics
Stanford University
Stanford, CA 94305, USA.
clausend@stanford.edu

## Abstract

With the dramatic growth of scientific publishing, Information Extraction (IE) systems are becoming an increasingly important tool for large scale data analysis. Hedge detection and uncertainty classification are important components of a high precision IE system. This paper describes a two part supervised system which classifies words as hedge or non-hedged and sentences as certain or uncertain in biomedical and Wikipedia data. In the first stage, our system trains a logistic regression classifier to detect hedges based on lexical and Part-of-Speech collocation features. In the second stage, we use the output of the hedge classifier to generate sentence level features based on the number of hedge cues, the identity of hedge cues, and a Bag-of-Words feature vector to train a logistic regression classifier for sentence level uncertainty. With the resulting classification, an IE system can then discard facts and relations extracted from these sentences or treat them as appropriately doubtful. We present results for in domain training and testing and cross domain training and testing based on a simple union of training sets.

## 1 Introduction

With the rapid increase in domain specific (biomedical) and domain general (WWW) text collections information extraction is an increasingly important tool for making use of these data sets. In order to maximize the usefulness of extracted relations an Information Extraction (IE) system needs the ability to separate the factual and reliable relationships from the uncertain and unreliable relationships. Most work on this problem has focused on the task of hedge detection where the goal is to classify a span of text as hedged or as non-hedged with the goal of facilitating sentence level classification of certain or uncertain. Much of the work was conducted within the framework of the BioNLP 2009 shared task sub task on uncertainty detection focusing on biomedical datasets (Kim et al., 2009) motivating further work in the biomedical NLP field (Aramaki et al., 2009; Conway et al., 2009). Other work has focused on creating annotated datasets from both a linguistically sophisticated perspective (Saurí and Pustejovsky, 2009) or from a language engineering perspective (Vincze et al., 2008).

Early work by Light et al. (2004) framed the task as determining the degree of speculation or uncertainty at the sentence level. The presence of a hedge cue, a phrase indicating that authors cannot back up their opinions or statements with facts, is a high precision feature of sentence level uncertainty. Other early work focused on semi-supervised learning due to a lack of annotated datasets (Medlock and Briscoe, 2007). Linguistically motivated approaches achieved a robust baseline on the sentence classification task (Kilicoglu and Bergler, 2008) although their training methods are hand tuned. Morante and Daelemans (2009) cast the problem as a sequence labeling task and show that performance is highly domain dependent and requires high precision hedge detection in order to perform the complex task of hedge scope labeling. Szarvas (2008) demonstrates that semi-supervised learning is even more effective with more labeled training data and sophisticated feature selection.

HedgeHunter is built to perform the CoNLL-2010 sentence uncertainty classification task. The task is a supervised learning task with training data drawn from Wikipedia and biomolecular articles and abstracts. Each training sentence is la-



*Proceedings of the Fourteenth Conference on Computational Natural Language Learning: Shared Task*, pages 120–125,
Uppsala, Sweden, 15-16 July 2010. 

beled as certain or uncertain and every hedge cue is also labeled. HedgeHunter separates the task into two stages: hedge detection and uncertainty classification, with the goal of producing an independent high precision hedge detection system for use in other tasks such as hedge scope detection. The system is designed to be expanded using semi-supervised learning although this is not implemented at this time. This paper will describe the hedge detection stage in Section 2 and the sentence classification stage in Section 3. Section 4 describes the evaluation of the system and Section 5 discusses the results. Section 6 discusses the results in a larger context and suggest future areas for improvement. Section 7 summarizes the conclusions.

## 2 Hedge Detection

Hedge detection is largely based on the identification of lexical items like *suggest* and *might* which indicate sentence level uncertainty. As a result, reasonable hedge detection in English can be accomplished by collecting a list of all lexical items that convey hedging. These include epistemic verbs (*may, might, could, should, can, ought to*), psychological verbs of perception, knowing or concluding (*seems, guess, suppose, hope, assume, speculate, estimate*), adverbs (*possibly, unlikely, probably, approximately*), adjectives (*quite, rare, apparent*) and many nouns. While some of these, especially the epistemic verbs, are often applied across domains to indicate hedge cues, many are unique to a particular domain. Further complicating hedge detection in English is the fact that the same word types occasionally have different, non-hedging uses.

The form of a hedge cue often acts as a high precision feature, whenever one is present in a sentence it is highly likely to be labeled as a hedge cue in the training set. Lexical hedge cues often vary from domain to domain and contain multiple words so non-lexical features are required for recognizing hedge cues robustly across domains although they are unlikely to provide a large benefit due to the largely lexical nature of hedges. As a result HedgeHunter uses both lexical and POS features for classification. Some hedges like *ought to* span multiple words so we also use positional features in order to capture multi-word hedges.

The hedge detection stage labels each word in a sentence independently. Labeling is done by logistic regression using Quasi-Newton minimization to set feature weights. This is a classification method that is both fast and robust for binary classification tasks like the one at hand. Features are drawn from the target word to be labeled and its context, the three words to the left and right of the target word. For the target word we extract features based on the word form, the word lemma and its POS as determined by a maximum entropy POS tagger trained on the PennTreebank implemented in Stanford JavaNLP. For the 6 words in the context window we also extract features based on the word, its lemma and its POS.

## 3 Uncertainty Classification

Uncertainty classification involves partitioning the set of sentences in a dataset into certain and uncertain classes. In most scientific writing sentences are generally certain so uncertain sentences are the minority class. This holds even more so for the Wikipedia dataset due to the method by which annotations were obtained and the encyclopedic nature of the dataset. Wikipedia hedge cues were identified by the presence of the *weasel word* tag which editors are allowed to append to spans of text in a Wikipedia article. These are often applied in a manner similar to hedge cues in the annotated biomedical datasets but they also focus on identifying non universal statements like those quantified by some or few. Due to the collaborative nature of Wikipedia, what qualifies as a *weasel word* varies greatly contributing to the increased variation in hedge cues in this dataset. Weasel words often get edited quickly so there are not many examples in the training set creating further difficulties.

The presence of one or more hedge cues in a sentence is a good indication that the sentence should be classified as uncertain, although as we will see in the results section, non-hedge features are also useful for this task. To capture this we extract features from each sentence including the number of hedge cues found by the hedge detection stage and the string value of the first four lexical hedge cues found in each sentence. To capture any other non-hedge words which may contribute to sentence level uncertainty, we also include BOW features based on vocabulary items with frequencies above the mean frequency in the corpus. This is achieved by creating binary features for the presence of every word in the vocab-

ulary.

Classification is again performed by a logistic regression using Quasi-Newton minimization. It should be stressed that all hedge related features used by the uncertainty classification stage are taken from the results of the hedge detection stage and not from the gold standard annotation data. This was done to allow the system to fold new unannotated sentences into the training set to perform semi-supervised learning. Time constraints and implementation difficulties prevented fully implementing this system component. Future work plans to extract high class conditional likelihood features from unannotated sentences, annotate the sentences based on treating these features as hedges, and retrain the hedge detection stage and uncertainty classification stage in an iterative manner to improve coverage.

## 4 Evaluation

The dataset provided for the CoNLL-2010 shared task consists of documents drawn from three separate domains. Two domains, biomedical abstracts and full articles, are relatively similar while the third, selected Wikipedia articles, differs considerably in both content and hedge cues for the reasons previously discussed. Overall the dataset contains 11,871 sentences from abstracts, 2,670 from full articles, and 11,111 from Wikipedia articles.

Performance for the hedge detection system was calculated at the word level while performance for the uncertainty classification stage was calculated at the sentence level using the classes of hedged and uncertain as the positive class for precision, recall and F1 statistics. We compare our hedge detection system to a state of the art system presented in Morante and Daelemans (2009) and trained on a dataset of 20,924 sentences drawn from clinical reports and biomedical abstracts and articles. The Morante system used 10 fold cross validation while our system randomly withholds 10 percent of the dataset for testing so our results may be viewed as less reliable. We do provide the first evaluation of one system on both domain specific and domain general datasets. Table 1 provides a breakdown of performance by system and dataset.

We evaluated the performance of the HedgeHunter system on the withheld training data including 5003 evaluation sentences from the biomedical domain and 9634 sentences from Wikipedia. For uncertainty classification we compare our system to the results from the CoNLL-2010 shared task comparing to the state of the art systems. For more details see the task description paper (Farkas et al., 2010). Table 2 summarizes the results for the closed domain training subtask. Table 3 summarizes the best performing systems in the Wikipedia and biomedical domain on the cross domain training subtask and compares to the HedgeHunter system.

| System | Precision | Recall | F1 |
|---|---|---|---|
| **Morante** | | | |
| Abstracts | .9081 | .7984 | .8477 |
| Articles | .7535 | .6818 | .7159 |
| Clinical | .8810 | .2751 | .41.92 |
| **HedgeHunter** | | | |
| Abstracts | .8758 | .5800 | .6979 |
| Articles | .8704 | .4052 | .5529 |
| Wikipedia | .5453 | .2434 | .3369 |
| All | .6289 | .3464 | .4467 |

Table 1: Hedge detection performance

| System | Precision | Recall | F1 |
|---|---|---|---|
| **Tang** | | | |
| Biomedical | .8503 | .8777 | .8636 |
| **Georgescul** | | | |
| Wikipedia | .7204 | .5166 | .6017 |
| **HedgeHunter** | | | |
| Biomedical | .7933 | .8063 | .7997 |
| Wikipedia | .7512 | .4203 | .5390 |

Table 2: Uncertainty classification performance closed

| System | Precision | Recall | F1 |
|---|---|---|---|
| **Li** | | | |
| Biomedical | .9040 | .8101 | .8545 |
| **Ji** | | | |
| Wikipedia | .6266 | .5528 | .5874 |
| **HedgeHunter** | | | |
| Biomedical | .7323 | .6405 | .6833 |
| Wikipedia | .7173 | .4168 | .5272 |

Table 3: Uncertainty classification performance cross

## 5 Results

The Hedge Detection stage performed slightly worse than the state of the art system. Although precision was comparable for biomedical articles and abstracts our system suffered from very low recall compared to the Morante system. The Morante system included chunk tagging as an approximation of syntactic constituency. Since many multi word hedge cues are constituents of high precision words and very frequent words (*ought to*) this constituency information likely boosts recall. Like the Morante system, HedgeHunter suffered a significant performance drop when tested across domains, although our system suffered more due to the greater difference in domains between biomedical and Wikipedia articles than between biomedical and clinical reports and due to the annotation standards for each dataset. HedgeHunter achieved better results on biomedical abstracts than the full articles due to higher recall based on the significantly larger dataset. Our system produced the worst performance on the Wikipedia data although this was mostly due to a drop in precision compared to the biomedical domain. This is in line with the drop in performance experienced by other systems outside of the biomedical domain and indicates that Wikipedia data is noisier than the peer reviewed articles that appear in the biomedical literature confirming our informal observations. Since the dataset has an overwhelming number of certain sentences and unhedged words, there is already a large bias towards those classes as evidenced by high overall classification accuracy (87% for certainty detection and 97% for hedge detection on all data) despite sometimes poor F1 scores for the minority classes. During development we experimented with SVMs for training but abandoned them due to longer training times and it is possible that we could improve the recall of our system by using a different classifier, a weaker prior or different parameters that allowed for more recall by paying less attention to class priors. We plan to expand our system using semi-supervised learning so it is not necessarily a bad thing to have high precision and low recall as this will allow us to expand our dataset with high quality sentences and by leveraging the vast amounts of unannotated data we should be able to overcome our low recall.

The uncertainty classification system performed robustly despite the relatively poor performance of the hedge detection classifier. The use of BOW features supplemented the low recall of the hedge detection stage while still relying on the hedge features when they were available as shown by feature analysis. We did not implement bi or trigram features although this would likely give a further boost in recall. Wikipedia data was still the worst performing domain although our cross domain system performed near the state of the art system with higher precision.

Overall our system produced a high precision hedge detection system for biomedical domain data which fed a high precision uncertainty classifier. Recall for the hedge detection stage was low overall but the use of BOW features for the uncertainty classification stage overcame this to a small degree. The amount of annotated training data has a significant impact on performance of the HedgeHunter system with more data increasing recall for the hedge detection task. For the sentence uncertainty task the system still performed acceptably on the Wikipedia data.

## 6 Discussion

HedgeHunter confirmed many of the findings of previous research. The most significant finding is that domain adaptation in the task of hedge detection is difficult. Most new domains contain different vocabulary and hedges tend to be highly lexicalized and subject to variation across domains. This is reinforced by feature analysis where the top weighted features for our hedge detection classifier were based on the word or its lemma and not on its POS. Once our system learns that a particular lexical item is a hedge it is easy enough to apply it precisely, the difficulty is getting the necessary training examples covering all the possible lexical hedge cues the system may encounter. The lexicon of hedge cues used in biomedical articles tends to be smaller so it is easier to get higher recall in this domain because the chance of seeing a particular hedge cue in training is increased. With the Wikipedia data, however, the set of hedge cues is more varied due to the informal nature of the articles. This makes it less likely that the hedge detection system will be exposed to a particular hedge in training.

One possible avenue for future work should consider using lexical resources like WordNet, measures of lexical similarity, or n-gram language models to provide backoff feature weights for un-

seen lexical items. This would increase the recall of the system despite the limited nature of annotated training sets by leveraging the lexical nature of hedges and their relatively closed class status.

We also found that the size of the training set matters significantly. Each domain employs a certain number of domain specific hedge cues along with domain general cues. While it is easy enough to learn the domain general cues, domain specific cues are difficult and can only be learned by seeing the specific lexical items to be learned. It is important that the training dataset include enough examples of all the lexical hedge cues for a specific domain if the system is to have decent recall. Even with thousands of sentences to train on, HedgeHunter had low recall presumably because there were still unseen lexical hedge cues in the test set. Future work should concentrate on methods of expanding the size of the training sets in order to cover a larger portion of the domain specific hedging vocabulary because it does not appear that there are good non-lexical features that are robust at detecting hedges across domains. This may include using lexical resources as described previously or by leveraging the high precision nature of hedge cues and the tendency for multiple cues to appear in the same sentence to perform semi-supervised learning.

This work also confirmed that hedge cues provide a very high precision feature for uncertainty classification. The highest weighed features for the classifier trained in the uncertainty classification stage were those that indicated the presence and number of lexical hedge cues. Contrary to some previous work which found that features counting the number of hedge cues did not improve performance, HedgeHunter found that the number of hedge cues was a strong feature with more hedge cues indicating an increased likelihood of being uncertain (Szarvas, 2008). It is largely a limitation of the task that we treat all uncertain sentences as equally uncertain. From a linguistic perspective a speaker uses multiple hedge cues to reinforce their uncertainty and our system seems to confirm that in terms of the likelihood of class membership even if the datasets do not encode the degree of uncertainty directly. Future work should focus on creating more sophisticated models of uncertainty that recognize the fact that it is at least a scalar phenomena and not a binary classification. Ideally a hedge detection and uncertainty quantification system would function to attach a probability to every fact or relation extracted from a sentence in an IE system determined in part by the hedging vocabulary used to express that fact or relation. This would yield a more nuanced view of how language conveys certainty and allow for interesting inference possibilities for systems leveraging the resulting IE system output.

One surprising finding was that uncertain sentences often contained multiple hedge cues, sometimes up to 4 or more. This is useful because it allows us to hypothesize that a sentence that is unannotated and has a high chance of being uncertain due to containing a hedge cue that we have seen in training, possibly contains other hedge cues that we have not seen. We can then use the large amounts of unannotated sentences that are available to extract n-gram features that have high uncertainty class conditional probability and add them to our training set with those features labeled as hedges as described in Medlock and Briscoe (2007). Because hedges are high precision features for uncertainty this should not hurt precision greatly. This allows us to increase the size of our training set substantially in order to expose our system to a greater variety of hedge cues in a semi-supervised manner. As with most semi-supervised systems we run the risk of drift resulting in a drop in precision. Future work will have to determine the correct balance between precision and recall, ideally by embedding this task within the larger IE framework to provide extrinsic evaluation

This work neglected to address the more difficult task of hedge scope detection. Determining hedge scope requires paring spans of sentences that fall within the hedge scope to a given hedge cue. Along with a move towards a scalar notion of uncertainty we should move towards a scope based instead of sentence based representation of uncertainty. Hedges take scope over subparts of a sentence so just because a relation occurs in the same sentence as a hedge cue does not mean that the given relation is hedged. It seems unnecessarily strict to ignore all relations or facts in a sentence just because it contains a hedge. Hedge detection is an important precursor to hedge scope detection. Without a high performing hedge detection system we cannot hope to link hedge cues with their respective scopes. This work hopes to produce a method for training such a hedge detection system for use as a component of a hedge

scope finding system.

This work also failed to integrate constituency or dependency features into either stage of the system. Dependencies encode important information and we plan to include features based on dependency relationships into future versions of the system. At the hedge detection stage it should improve recall by allowing the system to detect which multi word hedge cues are part of the same cue. At the uncertainty classification stage it should allow the extraction of multiword features not just based on n-gram frequency. For semi-supervised learning it should allow the system to more accurately annotated multi word features that have a high class conditional probability. This should be even more important when performing the task of hedge scope detection where scope is often delimitated at the phrase level and determining the dependency relations between words can capture this observation.

## 7 Conclusion

This work described HedgeHunter, a two stage hedge detection and uncertainty classification system. It confirmed the lexical nature of the hedge detection task, the importance of hedge cues to uncertainty classification and sharpened the need for large amounts of training data in order to achieve broad coverage. It highlights the issues involved in developing an open domain system by evaluating across very disparate datasets. It provides a framework that can be extended to semi-supervised learning in order to leverage large amounts of unannotated data to improve both in domain and cross domain performance.

## References


Eiji Aramaki, Yasuhide Miura, Masatsugu Tonoike, Tomoko Ohkuma, Hiroshi Mashuichi, and Kazuhiko Ohe. 2009. TEXT2TABLE: Medical Text Summarization System Based on Named Entity Recognition and Modality Identification. In *Proceedings of the BioNLP 2009 Workshop*, pages 185–192, Boulder, Colorado, June. Association for Computational Linguistics.

Mike Conway, Son Doan, and Nigel Collier. 2009. Using Hedges to Enhance a Disease Outbreak Report Text Mining System. In *Proceedings of the BioNLP 2009 Workshop*, pages 142–143, Boulder, Colorado, June. 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 (CoNLL-2010): Shared Task*, pages 1–12, Uppsala, Sweden, July. Association for Computational Linguistics.

Halil Kilicoglu and Sabine Bergler. 2008. Recognizing Speculative Language in Biomedical Research Articles: A Linguistically Motivated Perspective. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, pages 46–53, Columbus, Ohio, June. Association for Computational Linguistics.

Jin-Dong Kim, Tomoko Ohta, Sampo Pyysalo, Yoshinobu Kano, and Jun'ichi Tsujii. 2009. Overview of BioNLP'09 Shared Task on Event Extraction. In *Proceedings of the BioNLP 2009 Workshop Companion Volume for Shared Task*, pages 1–9, Boulder, Colorado, June. Association for Computational Linguistics.

Marc Light, Xin Ying Qiu, and Padmini Srinivasan. 2004. The Language of Bioscience: Facts, Speculations, and Statements in Between. In *Proc. of the HLT-NAACL 2004 Workshop: Biolink 2004, Linking Biological Literature, Ontologies and Databases*, pages 17–24.

Ben Medlock and Ted Briscoe. 2007. Weakly Supervised Learning for Hedge Classification in Scientific Literature. In *Proceedings of the ACL*, pages 992–999, Prague, Czech Republic, June.

Roser Morante and Walter Daelemans. 2009. Learning the Scope of Hedge Cues in Biomedical Texts. In *Proceedings of the BioNLP 2009 Workshop*, pages 28–36, Boulder, Colorado, June. Association for Computational Linguistics.

Roser Saurí and James Pustejovsky. 2009. FactBank: a corpus annotated with event factuality. *Language Resources and Evaluation*, 43(3):227–268.

György Szarvas. 2008. Hedge Classification in Biomedical Texts with a Weakly Supervised Selection of Keywords. In *Proceedings of ACL-08: HLT*, pages 281–289, Columbus, Ohio, June. Association for Computational Linguistics.

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(Suppl 11):S9.