Exploring hedge identification in biomedical literature

Ben Medlock

University of Cambridge, Computer Laboratory, William Gates Building, 15 JJ Thomson Avenue, Cambridge CB3 OFD, UK

Received 9 July 2007
Available online 11 January 2008

Abstract

We investigate automatic identification of speculative language, or ‘hedging’, in scientific literature from the biomedical domain. Our contributions include a precise description of the task including annotation guidelines, theoretical analysis and discussion. We show that good agreement can be achieved using our guidelines and present a publicly available benchmark dataset for the task. We argue for separation of the acquisition and classification phases in semi-supervised machine learning, and present a probabilistic acquisition model which is evaluated both theoretically and experimentally. We explore the impact of different sample representations on classification accuracy across the learning curve and demonstrate the effectiveness of using machine learning for the hedge identification task. Finally, we examine the errors made by our approach and point toward avenues for future research.

Keywords: Hedging; Classification; Natural language processing; Machine learning; Semi-supervised learning; Annotation; Agreement

1. Introduction

The automatic processing of scientific papers using natural language processing (NLP) and machine learning (ML) techniques is an increasingly important aspect of technical informatics. In the quest for a deeper machine-driven ‘understanding’ of the mass of scientific literature, a frequently occurring linguistic phenomenon that must be accounted for is the use of hedging to denote propositions of a speculative nature. As an example, consider the information conveyed by each of the following examples:

1. Our results prove that XfK89 inhibits Felin-9.
2. Our results suggest that XfK89 might inhibit Felin-9.

The second example contains a hedge, signalled by the use of suggest and might, which renders the proposition inhibit(XfK89 → Felin-9) speculative.

For an example of why analysis of hedging is important for automatic text processing, consider a system designed to identify and extract interactions between genetic entities in the biomedical domain. Case 1 above provides clear textual evidence of such an interaction and justifies extraction of inhibit(XfK89 → Felin-9), whereas case 2 provides only weak evidence and would probably not justify extraction.

Hedging occurs across the entire spectrum of scientific literature, though it is particularly common in the experimental natural sciences. In this study we consider the problem of learning to automatically classify sentences containing instances of hedging, given only a very limited amount of annotator-labeled ‘seed’ data. This falls within the semi-supervised ML framework, for which a range of techniques have been previously explored. The contributions of our work are as follows:

1. We provide a clear description of the problem of hedge identification and offer an improved and expanded set of annotation guidelines, along with illustrative examples, which as we demonstrate experimentally are sufficient to induce a high level of agreement between independent annotators.
2. We discuss the specificities of hedge identification as a semi-supervised ML task.
3. We argue for the separation of the acquisition and classification phases in semi-supervised learning.
4. We derive a probabilistic acquisition model and use it to motivate our approach.
5. We analyse the techniques presented both theoretically and experimentally, reporting promising results for the task on a new publicly available full-text dataset.¹

The work presented in this article draws and expands upon prior research presented in a recent paper by the author Medlock and Briscoe [22].

2. The task

Given a collection of sentences, \( S \), the task is to label each sentence as either speculative or non-speculative (spec or nspec henceforth). Specifically, \( S \) is to be partitioned into two disjoint sets, one representing sentences that contain some form of hedging, and the other representing those that do not.

It should be noted that by nature of the task definition, a speculative sentence may contain an arbitrary number of non-speculative assertions, leading to the question of whether hedge identification should be carried out at the granularity of assertions rather than sentences. While there is a strong argument in favour of this approach, it requires the identification of assertion boundaries, thus adding an extra level of complexity to all aspects of the task, from annotation to evaluation. In fact, even if the end goal is to label assertions, sentence level hedge identification can be viewed as an initial stage, after which potentially speculative sentences can be further examined to identify speculative constituents.

In an effort to further elucidate the nature of the task and to aid annotation, we have developed a new set of guidelines, building on the work of [19]. It is important to note that at least on a conceptual level, speculative assertions are not to be identified on the basis of the presence of certain designated hedge terms, rather the assessment is made based on a judgement of the author’s intended meaning, as revealed by the text.

We begin with the hedge definition given by [19] (item 1) and introduce a set of further guidelines to help elucidate various ‘grey areas’ and tighten the task specification.

The following ARE considered instances of hedging:

1. Any assertion relating to a result that does not necessarily follow from the work presented, but could be extrapolated from it [19], e.g.:

   \[ \text{This unusual substrate specificity may explain why Dronc is resistant to inhibition by the pan-caspase inhibitor.} \]

   Indeed, most mitochondria released all their cytochrome c, suggesting that an enzymatic transport mechanism is probably not involved.

   Our results provide the first direct evidence linking RAG1 and RSSs to a specific superfamily of DNA transposons and indicate that the V(D)J machinery evolved from transposons.

   A reduction of coverage could be the result of a reduction in dendrite outgrowth.

   Thus, nervy likely regulates multiple aspects of neuronal differentiation.

2. Relay of hedge made in previous work, e.g.:

   DI and Ser have been proposed to act redundantly in the sensory bristle lineage.

3. Statements of knowledge paucity, e.g.:

   How endocytosis of DI leads to the activation of N remains to be elucidated.

   Biochemical analysis of the ubiquitination events regulated by D-mib will be needed to further define the mechanism by which D-mib regulates the endocytosis of Ser in vivo.

   There is no clear evidence for cytochrome c release during apoptosis in Caenorhabditis elegans or Drosophila.

   There is no apparent need for cytochrome c release in C. elegans, since CED-4 does not require it to activate CED-3.

4. Speculative questioning, e.g.:

   A second important question is whether the roX genes have the same, overlapping or complementing functions.

5. Statement of speculative hypothesis, e.g.:

   To test whether the reported sea urchin sequences represent a true RAG1-like match, we cut off the ring finger motif and repeated the BLASTP search against all GenBank proteins.

6. Anaphoric hedge, e.g.:

   This hypothesis is supported by our finding that both pupariation rate and survival of…

   The rescue of the D-mib mutant phenotype by ectopic expression of N reveals strongly this interpretation.

The following are NOT considered instances of hedging:

1. Indication of experimentally observed non-universal behaviour, e.g.:

   Proteins with single BIR domains can also have functions in cell cycle regulation and cytokinesis.

   These results demonstrate that ADGF-A overexpression can partially rescue the effects of constitutively active Toll signalling in larvae.

   IAPs contain at least one BIR domain, and often a carboxy-terminal RING domain.

2. Confident assertion based on external work, e.g.:

   Two distinct E3 ubiquitin ligases have been shown to regulate DI signalling in Drosophila melanogaster.

3. Statement of existence of proposed alternatives, e.g.:

   Different models have been proposed to explain how endocytosis of the ligand, which removes the ligand from the cell surface, results in N receptor activation.

4. Confirmation of previous speculation, e.g.; *Here we show that the hemocytes (blood cells) are the main regulator of adenosine in the Drosophila larva, as was speculated previously for mammals.*

5. Confirmation of already firmly-stated conclusion *This conclusion is further supported by the even more efficient rescue achieved by...*

6. Negation of previous hedge *Although the adfg-a mutation leads to larval or pupal death, we have shown that this is not due to the adenosine or deoxyadenosine simply blocking cellular proliferation or survival, as the experiments in vitro would suggest.*

3. Related work

3.1. Hedge identification

While there is a certain amount of literature within the linguistics community on the use of hedging in scientific text, e.g. [13], there is little of direct relevance to the task of classifying speculative language from an NLP/ML perspective.

The most clearly relevant study is [19], where the hedge identification problem is introduced using examples drawn from the biomedical domain. They address the question of whether there is sufficient agreement among humans about what constitutes a speculative assertion to make the task viable from a computational perspective. At first they attempt to distinguish between two shades of speculation: strong and weak, but fail to garner sufficient agreement for such a distinction to be reliably annotated. However, they conclude that it is feasible to draw a reliable distinction between speculative and non-speculative sentences. They focus on introducing the problem, exploring annotation issues and outlining potential applications rather than on the specificities of the ML approach, though they do present some results using a manually crafted substring matching classifier and a supervised SVM on a collection of Medline abstracts. We will draw on this work throughout our presentation of the task.

Analysis of hedging in the context of citation function is carried out in [23], though they do not directly consider the task of hedge identification.

3.2. Semi-supervised learning

Recent years have witnessed a significant growth of research into semi-supervised ML techniques for NLP applications. Different approaches are often characterised as either multi- or single-view, where the former generate multiple ‘views’ on the data and perform mutual bootstrapping. This idea was formalised by [5] in their presentation of co-training which they show to be a powerful approach given the assumptions that: (1) each view is sufficient for classification and (2) the views are conditionally independent given the class label. These assumptions very rarely hold in real data, but co-training can still be effective under related but weaker conditions [1]. Co-training has also been used for named entity recognition (NER) [8], coreference resolution [26], text categorisation [27] and improving gene name data [32]. A number of researchers have proposed variants on the co-training idea. For example, rather than partitioning the feature space [11], generate multiple views by utilising two different machine learners, each of which is then used to bootstrap the other.

Conversely, single-view learning models operate without an explicit partition of the feature space. Perhaps the most well known of such approaches is expectation maximisation (EM), used by [27] in the context of learning from a combination of labeled and unlabeled data for text categorization. Others have proposed variations on the basic EM algorithm, for instance [26] present a two-tiered bootstrapping approach (EM-FS) in which EM is combined with a feature selection procedure to enhance its performance.

Another single-view algorithm occurring in the literature is called self-training, in which a labeled pool is incrementally enlarged with unlabeled samples for which the learner is most confident. Early work by [35] falls within this framework. He proposed a bootstrapping algorithm for learning new patterns given existing ones in an iterative process, utilising the redundancy inherent in the fact that the sense of a word is constrained by its current discourse usage (one sense per discourse), and also by local contextual cues. ‘Bagging’ and agreement are used to measure confidence on unlabeled samples in [3], and more recently [21] use self-training for improving parse reranking.

Other relevant recent work includes [36], in which random feature projection and a committee of SVM classifiers is used in a hybrid co/self-training strategy for semi-supervised relation classification and [7] where a graph based algorithm called label propagation is employed to perform semi-supervised relation extraction.

4. Motivation

In this section we provide some justification as to why the task of hedge identification is at least potentially tractable from a semi-supervised ML perspective.

The acquisition of new information about a particular target function from unlabeled data depends upon the existence of redundancy in the specification of the target function, even if the feature space cannot be explicitly partitioned into conditionally independent sets. This idea can be formalised as follows: given a particular feature, whose presence we will denote by $f_1$, a target function $Y$ and a learned hypothesis $H$, let us suppose that $f_1$ is a good indicator of a certain target function value $y$, e.g. $P(Y = y \mid f_1) \approx 1$. We will also assume that this is known to the learner, e.g. $P(H = y \mid f_1) = P(Y = y \mid f_1)$. To infer new information about $Y$ using an unlabeled source, there must exist some feature $f_2$, also a good indicator of $Y = y$, such that the following conditions hold:
1. $P(f_2 \mid f_1) > P(f_2)$
2. $P(f_2 \mid f_1, Y = y) < 1$

Condition 1 states that the features must not be negatively correlated, i.e., it must be possible to infer from instances containing $f_1$ that $f_2$ is also a good indicator of $Y = y$, while condition 2 states that the positive correlation between the two features, conditioned on the target class, must not be too tight, otherwise the learning process will grind to a halt. We can generalise from single features to ‘rules’ or ‘views’ that combine multiple features, but the same principles apply. Taken together, these conditions are a less precise, but for our task more intuitive version of the weak rule dependence condition of [1], which is itself a relaxed version of the conditional independence assumption of [5].

Analysis of the hedge identification task reveals potential redundancy of the above form that should be exploitable by a suitably chosen semi-supervised learner. We begin by assuming that features in our model are single terms, based on the intuition that many hedge cues are single terms (suggest, likely, etc.) and due to the success of ‘bag-of-words’ (BOW) representations in many learning tasks to date. Later, we will consider possible techniques for enriching the representation.

Consider again the example speculative sentence from earlier: “These results suggest that XfK89 might inhibit Felin-9.” Both suggest and might are hedge cues, and it is plausible to assume that they also occur within speculative sentences in other contexts, for instance “We suspect there might be an interaction between XfK89 and Felin-9.”. Now, for $f_1 = \text{suggest}$ and $f_2 = \text{might}$, we can examine the conditions specified above:

1. $P(\text{might}|\text{suggest}) > P(\text{might})$
2. $P(\text{might}|\text{suggest}, Y = \text{spec}) < 1$

The required values can be estimated from our data, yielding the following: $P(\text{might}|\text{suggest}) = 0.037$ and $P(\text{might}) = 0.012$. Given the (reasonable) approximation that suggest is always used as a hedge cue, $P(\text{might}|\text{suggest}) = P(\text{might}|\text{suggest}, Y = \text{spec})$ and both conditions hold.

While such evidence suggests that the task is feasible, there are a number of factors that make our formulation of hedge identification both interesting and challenging from a semi-supervised learning perspective. Firstly, each sample contains a potentially large number of irrelevant features, as hedging modifies the certainty with which an assertion is made, but in general does not modify the assertion itself, rendering most of the actual content of an assertion irrelevant. However, hedge cues come from a wide range of linguistic categories [14], mitigating against techniques such as traditional stopword filtering, and take many different forms. Consequently there are no obvious methods of removing irrelevant features without also losing potential hedge cues. Exacerbating this problem is the fact that speculative sentences may contain many non-speculative assertions, each potentially adding a large number of irrelevant features.

Such characteristics are in contrast to much previous work on semi-supervised learning, where for instance in the case of text categorization [5,27] almost all content terms are to some degree relevant, and irrelevant features are usually stopwords and can easily be filtered out. In the same vein, for the case of entity/relation extraction and classification [8,36,7] the context of the entity or entities in consideration provides a highly relevant feature space, and such studies are often set up such that only entities which fulfil some contextual criteria are considered [8].

Another interesting factor in our formulation of hedge identification that sets it apart from previous work is that the class of non-speculative sentences is defined on the basis of the absence of hedging, rather than on any positive characteristic. This makes it difficult to model the nspec class directly, and also hard to select a reliable set of nspec seed sentences, as by definition at the beginning of the learning cycle the learner has little knowledge about what a hedge looks like. The nspec seed problem is addressed in Section 8.5.

In this study we will develop a semi-supervised learning strategy based around the principle of iteratively predicting labels for unlabeled training samples. This is the basic paradigm for both co-training and self-training; however we will generalise by framing the task in terms of the acquisition of labeled training data, from which a supervised classifier can subsequently be trained to distinguish those sentences that contain hedging from those that don’t. It is our contention that there are good reasons for making the distinction between acquiring training data and classification, based on the observation that, while clearly related, the tasks are not the same. This distinction will become clearer in the next section when we develop a formal model for the learning procedure; however, using the arguments put forward in this discussion one can see informally where some of the distinctions lie. As we have seen, redundancy in the representation is crucial for acquiring new training samples; however this is not the case for classification. The aim of a classifier is to learn an accurate mapping between samples and target classes, and this does not require feature redundancy; in fact it is often beneficial to reduce redundancy by using features that specify the target classes more precisely. Given this insight, it may be advantageous to use different representations for the acquisition and classification phases, in addition to employing different model types.

A related, though somewhat orthogonal argument can be made from the point of view of data sparsity. At the start of the acquisition phase, there is only a very limited amount of training data (the seed samples), and a complex representation is likely to suffer excessively from issues of data sparsity. However, once a sufficiently large training set has been induced, this becomes much less of an issue, and a more complex representation might indeed be beneficial.
5. Acquisition

Having informally motivated the separation of the acquisition and classification phases in semi-supervised learning, we now present two variants of a simple probabilistic acquisition model, along with a brief description of an extant approach for purposes of experimental comparison.

5.1. Probabilistic acquisition model

Given:
- sample space $\mathcal{X}$
- set of target concept classes $\mathcal{Y} = \{y_1 \ldots y_N\}$
- target function $Y : \mathcal{X} \to \mathcal{Y}$
- set of seed samples for each class $S_1 \ldots S_N$ where $S_i \subset \mathcal{X}$ and $\forall X \in S_i[\{Y(x) = y_i\}$
- set of unlabeled samples $\mathcal{U} = \{x_1 \ldots x_k\}$

Aim: Infer a set of training samples $T_i$ for each concept class $y_i$ such that $\forall X \in T_i[Y(x) = y_i]$.

Now, it follows that $\forall X \in T_i[\{Y(x) = y_i\}$ is satisfied in the case that $\forall X \in T_i[P(y_i|x) = 1]$, which leads to a model in which $T_i$ is initialised to $S_i$ and then iteratively augmented with the unlabeled sample(s) for which the posterior probability of class membership is maximal. Formally:

At each iteration:

$$T_i \leftarrow x_j(\in \mathcal{U}) \quad \text{where} \quad j = \arg \max_j[P(y_i|x_j)]$$  

Expansion with Bayes’ Rule yields:

$$\arg \max_j[P(y_i|x_j)] = \arg \max_j[P(x_j|y_i) \cdot P(y_i)] / P(x_j)$$  

An interesting observation regarding Eq. (2) is the importance of the sample prior $P(x_j)$ in the denominator, often ignored for classification purposes because of its invariance to class. This provides a more formal qualification of the distinction between acquisition and classification.

At this point, we will explore two different ways of interpreting (2), the first by marginalising the sample prior and the second by estimating it directly. These interpretations will lead us to somewhat different approaches to the acquisition task.

5.1.1. Interpretation 1: marginalised sample prior (MSP)

Under this interpretation, we expand (2) by marginalising over the classes in the sample prior (denominator):

$$\arg \max_j\left[\frac{P(x_j|y_j) \cdot P(y_j)}{\sum_{i=1}^{N} P(y_i)P(x_j|y_i)}\right]$$  

leaving the class priors and class-conditioned likelihoods to be estimated from the data, which can be made tractable under limited dependence assumptions. The class priors can be estimated based on the relative distribution sizes derived from the current training sets:

$$P(y_i) = \frac{|T_i|}{\sum_i|T_i|}$$  

where $|S|$ is the number of samples in training set $S$. If we assume feature independence we can simplify the class-conditioned likelihood in the well known manner:

$$P(x_j|y_i) = \prod_k P(x_{jk}|y_i)$$  

and then estimate the likelihood for each feature:

$$P(x_{jk}|y_i) = \frac{zP(x_{jk}) + f(x_{jk}, T_i)}{zP(x_{jk}) + |T_i|}$$  

where $f(x,S)$ is the number of samples in training set $S$ in which feature $x$ is present, and $z$ is a universal smoothing constant, scaled by the class prior. This scaling is motivated by the principle that without knowledge of the true distribution of a particular feature it makes sense to include knowledge of the class distribution in the smoothing mechanism. Smoothing is particularly important in the early stages of the learning process when the amount of training data is severely limited resulting in unreliable frequency estimates.

5.1.2. Interpretation 2: pointwise mutual information (PMI)

Under this interpretation, we retain the sample prior from (2) in its un marginalised form and eliminate the class prior $P(y_i)$ from the numerator owing to its invariance with respect to the sample. Then, by taking the log of the result we derive the expression for the pointwise mutual information (PMI) [10,30] between the sample and the class:

$$\arg \max_j\left[\frac{P(x_j|y_j) \cdot P(y_j)}{P(x_j)}\right] \propto \arg \max_j\left[\log\frac{P(x_j|y_j)}{P(x_j)}\right] \propto \arg \max_j[\text{PMI}(x_j, y_j)]$$  

Thus we see that sample selection by maximal class posterior is equivalent to selection by maximal PMI. Under the feature independence assumption, the sample prior can be factorised into a product of its individual feature priors in a similar manner to the factorisation of the class-conditioned likelihood (5). Rearrangement yields:

$$\arg \max_j\left[\log\frac{P(x_j|y_j)}{P(x_j)}\right] = \arg \max_j\left[\log \prod_k P(x_{jk}|y_j) / P(x_{jk})\right]$$  

$$= \arg \max_j\left[\log \prod_k P(x_{jk}|y_j) / P(x_{jk})\right]$$  

$$= \arg \max_j\left[\sum_k \log P(x_{jk}|y_j) / P(x_{jk})\right]$$  

$$= \arg \max_j\left[\sum_k \text{PMI}(x_{jk}, y_j)\right]$$  

We are left with a summation over the PMI values for the individual features within each sample. $P(x_{jk}|y_j)$ (the per-feature class-conditioned likelihood) is estimated by (6) and we use a similar estimate for the feature prior:
\[ P(x_k) = \frac{\alpha + f(x_k, T)}{\alpha + |T|} \]  

(9)

where \( T \) represents the set containing all the training data, both labeled and unlabeled. This calculation is both efficient and highly perspicuous as the contributions of individual features are simply added together. However, the great advantage of this approach is that it allows (in principle) sample ranking proportional to \( P(\text{spec} | x_k) \) without requiring an estimate of \( P(\text{nspec} | x_k) \), and thus avoiding the potentially problematic generation of \( \text{nspec} \) seeds (Section 8.5).

The acquisition models we have presented (especially the MSP interpretation) are related to the multinomial Naive Bayesian models occurring throughout the NLP and machine learning literature, e.g. [20,9,29], usually in the context of classification. As far as we are aware, this is the first study to specifically examine the data acquisition task in a probabilistic framework.

The smoothing technique we describe is based on Laplacian smoothing [20] where a small value (often 1) is added to each feature count, though the prior-based scaling is a novel addition. Alternative techniques for smoothing the Naive Bayesian parameter estimates include back-off and interpolation [16,28]. Further exploration is a potential avenue for future research.

5.2. Acquisition for hedge identification

We will now consider how to apply the acquisition models to the hedge identification task. As discussed earlier, the speculative/non-speculative distinction hinges on the presence or absence of a few hedge cues within the sentence. Working on this premise, all features are ranked according to their probability of ‘hedge cue-ness’, given by the \( \text{spec} \) feature posterior in the MSP interpretation:

\[ P(\text{spec} | x_k) = \frac{P(x_k | \text{spec}) \cdot P(\text{spec})}{P(\text{spec})P(x_k | \text{spec}) + P(\text{nspec})P(x_k | \text{nspec})} \]  

(10)

which can be computed directly using (4) and (6), and the feature-class PMI in the PMI interpretation:

\[ \text{PMI}(\text{spec} | x_k) = \log \frac{P(x_k | \text{spec})}{P(x_k)} \]  

(11)

computed using (6) and (9).

The \( m \) most probable features are then selected from each sentence to compute (2) and the rest are ignored. This has the dual benefit of removing irrelevant features and also potentially reducing dependence between features, as the selected features will often be non-local and thus not too tightly correlated.

Note that this idea differs from traditional feature selection in two important ways:

1. Only features indicative of the \( \text{spec} \) class are retained, or to put it another way, \( \text{nspec} \) class membership is inferred from the absence of strong \( \text{spec} \) features.

2. Feature selection in this context is \textit{not} a preprocessing step. The classes are not re-modelled after selection; rather the original estimates are used. This has the effect of heavily skewing the posterior estimates in favour of the \( \text{spec} \) class, but this is acceptable for ranking purposes.

Of course, this ‘one-sided’ feature selection technique could be carried out prior to class estimation as a preprocessing step; however, we would not expect this to be effective, as the \( \text{nspec} \) class would then be severely misrepresented, and the \( \text{spec} \) estimates would suffer accordingly. Later we demonstrate this to be the case experimentally (Section 9.1).

5.3. Comparison—SVM Committee acquisition model

To provide a comparison for the acquisition models presented here, we also implement a more traditional acquisition procedure drawn from the semi-supervised learning literature. At each iteration a \textit{committee} of five support vector machine (SVM) classifiers is trained on randomly generated overlapping subsets of the training data and their cumulative confidence used to select items for augmenting the labeled training data. For more detailed description of this approach see [3,36].

6. Classification

The acquisition procedure returns a labeled dataset for each class, from which a classifier can be trained to identify sentences containing hedges. We experiment with two separate classification approaches:

- \textit{Support Vector Machine}: we use the SVM implementation in Joachims’ SVM\textsc{light} [15], due to its efficiency and proven accuracy in classification tasks [33,18].
- \textit{Probabilistic}: we derive a simple probabilistic classifier using the estimates from the acquisition model:

\[ x_i \rightarrow \text{spec} \quad \text{if} \quad P(\text{spec} | x_i) > \sigma \]  

(12)

where \( \sigma \) is an arbitrary threshold used to control the precision/recall balance. Under the second acquisition model interpretation we use the PMI estimates instead of the posterior:

\[ x_i \rightarrow \text{spec} \quad \text{if} \quad \text{PMI}(x_i, \text{spec}) > \sigma \]  

(13)

As a baseline, we also use the substring matching classifier of [19], which labels a sentence as \( \text{spec} \) if it contains one or more of the following: suggest, potential, likely, may, at least, in part, possibl, further investigation, unlikely, putative, insights, point toward, promise, propose.
7. Sample representation

We use the single term bag-of-words (BOW) sample representation as a baseline, and additionally examine the following methods of enriching the representation:

7.1. Part-of-speech tagging

There is theoretical motivation for using part-of-speech (PoS) tags to augment the BOW representation from the point of view of sense disambiguation. Terms such as potential are highly indicative of hedging when used in adjectival form but less so in nominal form. For example, consider the following sentence:

An overview of the local backbone potential is shown in Fig. 5.

In this context the nominal use of potential does not indicate hedging and the sentence is quite clearly non-speculative. Contrast with:

The transient cmg transcription in midgut and Malpighian tubules suggests a potential function in cell junction formation and in epithelial tissue patterning.

where potential is quite clearly used as a hedge. PoS information will not always help though; consider a further example:

The UAS-brk transgene was amplified from potential mutants by PCR and sequenced.

It is clear (at least in the authors’ opinion) that this sentence is non-speculative and that the adjectival use of potential is not a hedge but rather part of the experimental description.

To create the new representation we suffixed each term with its respective part-of-speech tag using the RASP PoS component, based on a sequential HMM tagger and the CLAWS2 tagset.2

7.2. Stemming

We also experimented with stemming (using the Porter stemmer3). The motivation for stemming in hedge identification is that distinct morphological forms of (particularly verbal) hedge cues are often used to convey the same semantics, for instance:

Thus these data suggest that dpp signalling interacts with the retinal determination pathway.

and

There is a certain amount of evidence suggesting that dpp signalling interacts with the retinal determination pathway.

both convey clear speculation through variants of the root verb suggest. Verbal forms of nominal hedge cues (and vice-versa) and collapsed in this representation, so for instance hypothesis and hypothesize are both reduced to hypothesis. We generate representations that use both stemming on its own (including case normalisation) and in combination with PoS tagging.

7.3. Bigrams

There are many instances where combinations of terms represent more reliable hedge cues than just single terms. For instance, consider the following sentence:

In addition several studies indicate that in mammals the Rel proteins could probably be involved in CNS processes such as neuronal development and synaptic plasticity.

Analysis reveals that ‘indicate that’ is a fairly reliable hedge cue, whereas indicate on its own is not, because of instances such as the following:

In the row marked dgqa the stippled exons indicate regions that are not found in the dgqa cDNAs identified by us.

Notice that it is not the presence of indicate and that in the same sentence that is important here (both sentences fulfil that criterion), rather it is their adjacency. This suggests that bigram features may be useful, and could potentially enhance the sample representation. Using bigrams results in a well known explosion of the feature space \(O(n)\) and this often prohibits their usefulness due to issues of data sparsity. However the hedge identification problem possesses some characteristics that work to its advantage in this regard. Because the number of hedge cues is relatively small, the explosion occurs mostly in the space of irrelevant features, and with a reasonably large amount of data we would expect to see the same hedge constructions occurring often enough to yield at least fairly reliable statistics. Almost all of the research into complex feature generation has concluded that improvements are only gained through combining bigrams and single terms [31,24,4]. This has the added advantage that in our case such a scheme is guaranteed to at least retain the redundancy of the original representation, and almost certainly to increase it.

We include all adjacent bigrams and allow the learning models to select (explicitly or implicitly) the relevant ones.

8. Experimental setup

8.1. Data

For our experiments, we used an archive of 5579 full-text papers from the functional genomics literature relating

---

2 www.comp.lancs.ac.uk/ucrel/claws2tags.html.
3 http://www.tartarus.org/martin/PorterStemmer.
to *D. melanogaster* (the fruit fly). The papers were converted to XML and linguistically processed using the RASP toolkit\(^4\) [6]. We annotated six of the papers to form a test set with a total of 380 *spec* sentences and 1157 *nspec* sentences, and randomly selected 300,000 sentences from the remaining papers as training data for the semi-supervised learner. The unlabeled sentences were chosen under the constraints that they must be at least 10 words long and contain a main verb.

### 8.2. Annotation and agreement

Two separate annotators were commissioned to label the sentences in the test set, firstly one of the authors and secondly a domain expert with no prior input into the guideline development process. The two annotators labeled the data independently using the guidelines outlined in Section 2. Relative \(F_{rel}^{j}()\) and Cohen’s Kappa \((\kappa)\) were then used to quantify the level of agreement. For brevity we refer the reader to [2] and [12] for formulation and discussion of \(\kappa\) and \(F_{rel}^{j}\), respectively.

The two metrics are based on different assumptions about the nature of the annotation task. \(F_{rel}^{j}\) is founded on the premise that the task is to recognise and label *spec* sentences from within a background population, and does not explicitly model agreement on *nspec* instances. It ranges from 0 (no agreement) to 1 (no disagreement). Conversely, \(\kappa\) gives explicit credit for agreement on both *spec* and *nspec* instances. The observed agreement is then corrected for ‘chance agreement’, yielding a metric that ranges between \(-1\) and 1. Given our definition of hedge identification and assessing the manner in which the annotation was carried out, we suggest that the founding assumption of \(F_{rel}^{j}\) fits the nature of the task better than that of \(\kappa\).

Following initial agreement calculation, the instances of disagreement were examined. It turned out that the large majority of cases of disagreement were due to negligence on behalf of one or other of the annotators (i.e. cases of clear hedging that were missed), and that the cases of genuine disagreement were actually quite rare. New labelings were then created with the negligent disagreements corrected, resulting in significantly higher agreement scores. Values for the original and negligence-corrected labelings are reported in Table 1.

Annotator conferral violates the fundamental assumption of annotator independence, and so the latter agreement scores do not represent the true level of agreement; however, it is reasonable to conclude that the actual agreement is approximately lower bounded by the initial values and upper bounded by the latter values. In fact even the lower bound is well within the range usually accepted as representing ‘good’ agreement, and thus we are confident in accepting human annotation as a gold-standard for the hedge identification task. For our experiments, we use the labeling of the genetics expert, corrected for negligent instances.

### 8.3. Methodology

In each of our experiments we use the following method:

**Given:**

- seed sets \(S_{spec}\) and \(S_{nspec}\) (*nspec* not required for PMI acquisition)
- unlabeled sample set \(U\)

1. Initialise training data: \(T_{spec} ← S_{spec}\) and \(T_{nspec} ← S_{nspec}\)
2. Iterate:

   - Order \(U\) by ranking metric:
     - \(P(spec|x)\) for MSP interpretation (Section 5.1.1)
     - \(PMI(x, spec)\) for PMI interpretation (Section 5.1.2)
     - Cumulative SVM confidence for committee-based approach (Section 5.3)
   - \(T_{spec} ←\) most probable batch
   - \(T_{nspec} ←\) least probable batch
   - Train classifier using \(T_{spec}\) and \(T_{nspec}\)
   - Compute *spec* recall/precision BEP (break-even point) on the test data

The batch size for each iteration is set to 0.001 \(*|U|\). After each learning iteration, we compute the precision/recall BEP for the *spec* class using both classifiers trained on the current labeled data. We use BEP because it helps to mitigate against misleading results due to discrepancies in classification threshold placement. Disadvantageously, BEP does not measure a classifier’s performance across the whole of the recall/precision spectrum (as can be obtained, for instance, from receiver-operating characteristic (ROC) curves), but for our purposes it provides a clear, abstracted overview of a classifier’s accuracy given a particular training set.

### 8.4. Parameter setting

The training and classification models we have presented require the setting of two parameters: the smoothing parameter \(\alpha\) and the number of features per sample \(m\). Analysis of the effect of varying \(\alpha\) on feature ranking reveals that when \(\alpha = 0\), low frequency terms with spurious class correlation dominate and as \(\alpha\) increases, high frequency terms become increasingly dominant, eventually smoothing away genuine low-to-mid frequency correlations. This effect is illustrated in Table 2, and from this

---

4 www.informatics.susx.ac.uk/research/nlp/rasp.

---

### Table 1

<table>
<thead>
<tr>
<th>Agreement scores</th>
<th>(F_{rel}^{j})</th>
<th>(\kappa)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original</td>
<td>0.8293</td>
<td>0.9336</td>
</tr>
<tr>
<td>Corrected</td>
<td>0.9652</td>
<td>0.9848</td>
</tr>
</tbody>
</table>
analysis we chose $z = 5$ as an appropriate level of smoothing for the MSP interpretation.

In Section 5.1 we introduced the idea of prior-scaled smoothing. Under the PMI interpretation it plays a crucial role in the emergence of useful feature-class correlations. Table 3 demonstrates this phenomenon. To obtain these results we also used the following non-scaled formulation of the per-feature class-conditional likelihood:

$$P(x_k|y_i) = \frac{z + f(x_k, T_i)}{z + |T_i|}$$

(14)

Few, if any useful correlations emerge when using 14; however, when using scaled smoothing, genuine correlations do emerge. Note that we must use a higher value of $z$ under the PMI interpretation. The reason for the effectiveness of scaled smoothing is that the amount of smoothing is related to the current estimate of the focus class prior in relation to the whole body of training data, which at the start of the learning process is low. This encourages variation in the per-feature class-conditional estimate, while utilising the higher $z$ value to dampen the effect of low frequency terms in the feature prior.

We use $m = 5$ for both acquisition model interpretations, based on the intuition that five is a rough upper bound on the number of hedge cue features likely to occur in any one sentence. We do not contend that these are necessarily optimal parameter values for the task, rather that they are sensible.

We use the linear kernel for SVMlight with the default setting for the regularization parameter C. We construct binary valued, $L_2$-normalised (unit length) input vectors to represent each sentence, as this resulted in better performance than using frequency-based weights and concords with our presence/absence feature estimates.

### 8.5. Seed generation

The models we have presented require a set of seeds for each class. To generate seeds for the spec class, we extracted all sentences from $U$ containing either (or both) of the terms suggest or likely, as these are very good (though not perfect) hedge cues, yielding 6423 spec seeds. Generating seeds for nspec is much more difficult, as integrity requires the absence of hedge cues, and this cannot be done automatically. Thus, we used the following procedure to obtain a set of nspec seeds:

#### Table 3

Features ranked by PMI($x_i, \text{spec}$) with and without scaling

<table>
<thead>
<tr>
<th>Rank</th>
<th>$x = 5$</th>
<th>$x = 100$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Scaled</td>
<td>Non-scaled</td>
</tr>
<tr>
<td>1</td>
<td>suggest</td>
<td>interacts with</td>
</tr>
<tr>
<td>2</td>
<td>likely</td>
<td>Nonautonomous</td>
</tr>
<tr>
<td>3</td>
<td>aDNA</td>
<td>Taken</td>
</tr>
<tr>
<td>4</td>
<td>EGFP</td>
<td>Together</td>
</tr>
<tr>
<td>5</td>
<td>learn</td>
<td>findings</td>
</tr>
<tr>
<td>6</td>
<td>Adelsberger</td>
<td>observations</td>
</tr>
<tr>
<td>7</td>
<td>Ubxxx</td>
<td>These</td>
</tr>
<tr>
<td>8</td>
<td>xCopenxD</td>
<td>polytypic</td>
</tr>
<tr>
<td>9</td>
<td>FNIII</td>
<td>hairing</td>
</tr>
<tr>
<td>10</td>
<td>Wingrove</td>
<td>variegation</td>
</tr>
<tr>
<td>11</td>
<td>dLgIPAR</td>
<td>together</td>
</tr>
<tr>
<td>12</td>
<td>earlystage</td>
<td>txBrotein</td>
</tr>
<tr>
<td>13</td>
<td>CRN</td>
<td>Dor-dependent</td>
</tr>
<tr>
<td>14</td>
<td>Pfliciparum</td>
<td>iicated</td>
</tr>
<tr>
<td>15</td>
<td>gel-like</td>
<td>peptidelipid</td>
</tr>
<tr>
<td>16</td>
<td>peroxisosmalXd</td>
<td>xBlighty</td>
</tr>
<tr>
<td>17</td>
<td>polyQ-</td>
<td>expanded</td>
</tr>
<tr>
<td>18</td>
<td>misannotated</td>
<td>Keen</td>
</tr>
<tr>
<td>19</td>
<td>ratiosx</td>
<td>CxBBA</td>
</tr>
<tr>
<td>20</td>
<td>GENERAL</td>
<td>C&gt;CxBBA</td>
</tr>
</tbody>
</table>

### Table 2

Features ranked by $P(\text{spec}|x_k)$ for varying $z$

<table>
<thead>
<tr>
<th>Rank</th>
<th>$z = 0$</th>
<th>$z = 1$</th>
<th>$z = 5$</th>
<th>$z = 100$</th>
<th>$z = 500$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>interacts with</td>
<td>suggest</td>
<td>suggest</td>
<td>suggest</td>
<td>suggest</td>
</tr>
<tr>
<td>2</td>
<td>TAFb</td>
<td>likely</td>
<td>likely</td>
<td>likely</td>
<td>likely</td>
</tr>
<tr>
<td>3</td>
<td>sexta</td>
<td>may</td>
<td>may</td>
<td>may</td>
<td>may</td>
</tr>
<tr>
<td>4</td>
<td>CRYs</td>
<td>might</td>
<td>might</td>
<td>These</td>
<td>These</td>
</tr>
<tr>
<td>5</td>
<td>DsRed</td>
<td>seems</td>
<td>results</td>
<td>results</td>
<td>results</td>
</tr>
<tr>
<td>6</td>
<td>Nonautonomous</td>
<td>suggests</td>
<td>Taken</td>
<td>might</td>
<td>that</td>
</tr>
<tr>
<td>7</td>
<td>arva</td>
<td>probably</td>
<td>suggests</td>
<td>observations</td>
<td>be</td>
</tr>
<tr>
<td>8</td>
<td>inter-homologue</td>
<td>suggesting</td>
<td>probably</td>
<td>Taken</td>
<td>data</td>
</tr>
<tr>
<td>9</td>
<td>Mohanty</td>
<td>possibly</td>
<td>Together</td>
<td>findings</td>
<td>it</td>
</tr>
<tr>
<td>10</td>
<td>meld</td>
<td>suggested</td>
<td>Our</td>
<td>Our</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>aDNA</td>
<td>Taken</td>
<td>possibly</td>
<td>seems</td>
<td>observations</td>
</tr>
<tr>
<td>12</td>
<td>Deer</td>
<td>unlikely</td>
<td>suggested</td>
<td>together</td>
<td>role</td>
</tr>
<tr>
<td>13</td>
<td>Borel</td>
<td>Together</td>
<td>findings</td>
<td>Together</td>
<td>most</td>
</tr>
<tr>
<td>14</td>
<td>substripe</td>
<td>physiology</td>
<td>observations</td>
<td>role</td>
<td>these</td>
</tr>
<tr>
<td>15</td>
<td>Failing</td>
<td>modulated</td>
<td>Given</td>
<td>that</td>
<td>together</td>
</tr>
<tr>
<td>16</td>
<td>uncommitted</td>
<td>reflecting</td>
<td>unlikely</td>
<td>be</td>
<td>might</td>
</tr>
<tr>
<td>17</td>
<td>distxAxFact</td>
<td>destruction</td>
<td>These</td>
<td>it</td>
<td>findings</td>
</tr>
<tr>
<td>18</td>
<td>descend</td>
<td>cooperative</td>
<td>reflect</td>
<td>strongly</td>
<td>more</td>
</tr>
<tr>
<td>19</td>
<td>excretions</td>
<td>Preliminary</td>
<td>results</td>
<td>most</td>
<td>function</td>
</tr>
<tr>
<td>20</td>
<td>actinC</td>
<td>outcome</td>
<td>Our</td>
<td>data</td>
<td>is</td>
</tr>
</tbody>
</table>
1. Create initial $S_{\text{nspec}}$ by sampling randomly from $U$.
2. Manually remove more ‘obvious’ speculative sentences using pattern matching.
3. Iterate:
   - Order $S_{\text{nspec}}$ by $P(\text{spec}|x_i)$ using estimates from $S_{\text{spec}}$ and current $S_{\text{nspec}}$.
   - Examine most probable sentences and remove speculative instances.

We started with 8830 sentences and after a couple of hours work reduced this down to a (still somewhat noisy) $nspec$ seed set of 7541 sentences.

9. Experimental results

In this section we present results for the acquisition and classification models introduced in this study using the experimental setup described above.

Firstly, we compare various configurations of the probabilistic (MSP interpretation) and SVM committee-based acquisition models with the baseline classifier. Fig. 1 plots accuracy as a function of the acquisition iteration. After 150 iterations, all of the semi-supervised approaches are significantly more accurate than the baseline classifier according to a binomial sign test ($p < 0.01$), though there is clearly still much room for improvement. The baseline classifier achieves a BEP of 0.60 while both classifiers reach approximately 0.76 BEP using the probabilistic acquisition model, with the SVM performing slightly better overall. The weakest combination is the SVM committee-based acquisition model with an SVM classifier. Interestingly though, the probabilistic classifier with the SVM committee-based acquisition model performs competitively with the other approaches. Overall, these results favour a framework in which the acquisition and classification phases are carried out by different models.

The PMI acquisition model did not fare as well as the MSP interpretation. Fig. 2 plots the learning curve for the PMI acquisition model and derived classifier, using the smoothing parameter values $\alpha = 25$ and 75. For clarity we omit alternative configurations, as they yielded similar results. An explanation for the weakness of this model follows from an examination of its theoretical properties. Samples are chosen on the basis of PMI, given by $\log \frac{P(x_i | y_i)}{P(x_i)}$. While our estimate of the sample prior $P(x_i)$ is quite reliable (given the independence assumption), the class-conditional likelihood estimate $P(x_i | y_i)$ is unreliable at the beginning of the learning cycle as it is estimated only from the seed data. In particular, many genuine feature-class correlations are weak, and even with the scaled smoothing they tend to be ‘drowned out’ when competing with prior estimates from the entire training corpus. Conversely, under the MSP interpretation, both the class conditional and prior estimates are weak at the start of the learning cycle which allows genuine spec correlations to emerge much more readily as the marginalised prior estimates are drawn only from the augmented $spec$ and $nspec$ seed data.
9.1. Comparison with traditional feature selection

The results presented in Fig. 1 for the probabilistic classifier use the one-sided feature selection technique outlined in Section 5.2, while the SVM results are obtained without feature selection. Fig. 3 plots the results of experiments in which we carry out one-sided feature selection for both classifiers as a preprocessing step, in order to test its expected ineffectiveness when used in the traditional way. As anticipated, both classifiers perform worse in this scenario, with a particularly dramatic decrease in accuracy for the SVM. We attribute this phenomenon to the fact that in the one-sided feature selection scenario, the SVM must discriminate between classes that are both represented by features indicative of the spec class; a task at which it is intuitively destined to fail. We also carried out experiments...
(not reported here) using traditional feature selection on both classes (with $\chi^2_{\text{max}}$) for the SVM classifier, and this resulted in poorer performance than using all features.

9.2. Seed refinement

Given the results obtained using the seed data described in Section 8.5, we wanted to measure the accuracy gained by additional manual improvement of the $nspc$ seed data. Our aim was to examine the hypothesis that even a relatively small number of spurious $spc$ sentences amongst the $nspc$ seeds could cause the learner to erroneously disregard some of the more subtle $spc$ class-feature correlations and thus significantly degrade the diversity of the induced training data.

We again employed the iterative refinement method described above (Section 8.5), and additionally used the substring classifier of [19] to extract a list of further potentially speculative sentences from which we removed the genuinely speculative ones and returned the rest to the $nspc$ seed set.

We spent around 2–3 hours refining the $nspc$ seeds and succeeded in removing 260 spurious instances, yielding a new $nspc$ seed set of 7281 $nspc$ sentences. Running the probabilistic acquisition model and SVM classifier using the new $nspc$ seeds yielded the results shown in Fig. 4. There is an improvement of around 1–2% BEP.

9.3. Exploring $nspc$ acquisition

Examining the sentences chosen by our acquisition model for augmenting the $nspc$ training data reveals a noticeable homogeneity of form. Fig. 5 shows a batch of sentences chosen by the learner for the $nspc$ class on the 15th training iteration. Almost all of these are descriptions of experimental methodology, and as such exhibit certain common features, such as past tense predicate constructions. Adding increasing numbers of methodology sentences to the $nspc$ training data would appear less than optimal in terms of modelling the $nspc$ class, as there are many $nspc$ sentences in other sections.

To compensate for this acquisition bias we explored a number of methods of guiding the learner to choose a wider range of $nspc$ sentence types:

1. Remove a significant proportion (around 25,000) of the methodology sentences from the unlabeled pool using source paper markup.
2. Filter the unlabeled pool to contain only sentences identified by the source paper markup as coming from one of the following sections: Summary, Introduction, Discussion, Results and Conclusions, leaving 108,694 sentences in the pool. These are the sections in which speculations are most likely to be made [23], and the idea is that the learner chooses $nspc$ sentences that are of a similar type to the $spc$ ones, thus giving the classifier more chance of discriminating between the difficult instances.

From Fig. 6 it can be seen that neither approach (designated ‘Filtered pool (1)’ and ‘Filtered pool (2)’, respectively) was able to provide a clear improvement over the original unfiltered pool. The danger of the second
approach is that the learner is also more likely to acquire spurious nspec sentences; it is also possible that in this scenario a significant proportion of useful spec sentences are removed from the pool, which may contribute to the decrease in performance.

We also tested a scenario in which the nspec training data is fixed in its initial seed state and only the spec training set is augmented, denoted by ‘Fixed nspec’. It is interesting that this has only a marginal negative impact on performance which suggests that a relatively small amount is learned about the nspec class through the acquisition process, beyond the information already contained in the seed data.

9.4. Enriched sample representations

9.4.1. PoS tagging

The results for the augmented PoS representation are given in Fig. 7. The addition of PoS tags does yield slightly better accuracy in later training iterations than the basic term-based representation, but the improvements are marginal and not statistically significant. In practice, the benefits derived from PoS tags in terms of word sense disambiguation are not as pronounced as theory might suggest. For example, earlier we argued that the term potential when used as an adjective is much more likely to represent hedging than when used as a nominal. While this is...
undoubtedly true, our test data contains no instances of the nominal form of potential, and both the spec and aspec sentences contain the same number of adjectival instances (five). Consider the following:

There was considerable excitement in the field when potential mammalian and Drosophila homologues for ced- and egl- were discovered.

The annotators decided that the use of potential in this instance did not represent an authorial hedge because potentiality is by necessity a property of homology. This exemplifies the notion that whether a particular term acts as a hedge cue is quite often a rather subtle function of its sense usage, in which case the distinctions may well not be captured by PoS tagging.

9.4.2. Stemming

Fig. 8 shows that the combined stemming/PoS representation follows much the same pattern as the original, which is unsurprising given that PoS tagging counteracts the generalisation achieved by stemming. For example, there are separate PoS tags for different verb tenses, which is precisely the sort of information that is discarded by stemming. The interesting case is when stemming is used alone. Over early training iterations, the accuracy of the classifier is significantly lower; however performance continues to improve in latter iterations, yielding a peak result of around 0.8 BEP. Carrying out a binomial sign test comparing the performance of the original and stemmed representations around their relative peaks (80 training iterations for the original representation and 120 for the stemmed variant) showed a weakly significant improvement ($p < 0.2$) for the stemmed representation.

9.4.3. Bigrams

We use the best performing stemmed single term representation and augment it with all adjacent bigrams. An example of a sentence and its representation is as follows:

Several lines of evidence suggest that upregulation of RD gene expression by dpp and ey is likely to account for the synergy that we have observed.

The results are shown in Fig. 9, and demonstrate that including bigrams yields a clear improvement in accuracy across most of the acquisition curve, with a new peak performance of around 0.82 BEP. According to the binomial sign test, this indicates a statistically significant improvement over both the original representation ($p < 0.01$) and the previous best performing stemmed representation ($p < 0.1$).

---

5 Biological homology refers to structural similarity resulting from shared ancestry, which cannot be established beyond question due to inherent lack of observability; thus is only ever ‘potential’.
Table 4 shows the 120 highest ranked features according to $P(spec|x_i)$ in the combined single term/bigram representation after 100 learning iterations. There are 13 single terms and 107 bigrams, and it is interesting to note that in some cases neither of the constituent single terms in a given bigram is a likely hedge cue while the combined feature clearly is; for instance ‘not_known’ (rank 112), which is a cue for the knowledge paucity hedge.

10. Error analysis

We examined the errors made by the SVM classifier after 100 iterations of the probabilistic acquisition model using the stem + adjacency bigram sample representation ($m = 5$, $\alpha = 5$). A BEP of 0.816 was obtained at this stage, equating to 310 correctly classified instances out of 380 for the spec class and 1087
out of 1157 for the nspec class (70 misclassified instances in each class).

A significant proportion (approx. 20%) of the missed spec instances were statements of knowledge paucity. These ranged from quite common forms, e.g.:

*The role of the roX genes and roX RNAs in this process is still unclear.*

...to more unusual variants, e.g.:

*This brings us to the largest of all mysteries, namely how the DCC is spread along the X chromosome.*

Such instances are further in construction from the spec seed sentences and thus somewhat harder to acquire training data for. A possible way of capturing these instances would be to include specific knowledge paucity seeds.

Some of the missed spec instances were due to cases where speculativity is indicated by a particular term, while the general construction of the sentence does not fit the usual spec mold. For example:

*We then tested the putative RNA-binding property of MOF directly using electromobility shift assays.*

In some cases, genuine hedge cues were not induced with enough certainty, leading to missed spec instances, for example:

*Invertebrates in vivo RAG-mediated transpositions are strongly suppressed, probably to minimize potential harm to genome function.*

The term *probably* is actually a fairly reliable hedge cue, but it only appears at rank 1268 in the list of features ranked according to $P(spec|x_k)$, estimated from the automatically acquired training data.

Quite a number of missed spec instances were just hard to classify, for example:

*Mutants that pupariated usually showed typical GFP expectoration indicating the presence of a high premorphic peak of ecdysteroids.*

It could certainly be argued that this is in fact an instance of observed non-universal behaviour, rather than a hedge. Another example is the following:

*Some of the intermediate stages of RAG evolution can be inferred from analysis of the sea urchin in which RAG-like proteins were recently observed, and from analysis of the lancelet starlet sea anemone and hydra genomes.*
This instance could be interpreted as a sort of ‘meta speculation’, stating that speculations about RAG evolution could be made from recent experimental findings. However, it is unclear as to whether this should constitute a hedge in itself.

The majority of false positives (nspec instances labeled as spec) were due to constructions that are hard to distinguish due to similarity of form, for example:

IAPs were first discovered in baculovirus but have since been shown to play a vital role in blocking apoptosis in Drosophila as well as in mammals.

Variants of the phrase ‘play a role’ are quite often used as hedge cues, and hence this instance looks like a hedge, though the annotators decided that in fact it is not.

Another example of confusion due to similarity of construction is the following:

Three Drosophila BIRPs have been shown to be inhibitors of apoptosis Diap Diap and Deterin.

The infinitive ‘to be’ and the verb be are in general quite reliable hedge cues (ranked 142 and 217, respectively) whereas in this instance they are not used to signal speculativity. This is also a potential indicator of the disadvantage of combining single terms and bigrams in terms of feature independence violation (though our results show that the benefits outweigh the disadvantages).

In some cases, the classifier actually identified spuriously labeled instances, for example the following were labeled by the annotators as nspec when they clearly contain hedges:

Caspases can also be activated with the aid of Apaf, which in turn appears to be regulated by cytochrome c and dATP.

and

Further insight into a possible mechanism for IAP function was recently gained when IAPs were observed to have ubiquitin ligase activity.

We found that around 10% of the false positives were actually due to spurious manual labeling.

11. Conclusions and future work

We have shown that semi-supervised ML is applicable to the problem of hedge identification and that a reasonable level of accuracy can be achieved. Our main contributions are:

- We derive two different interpretations of a simple but novel probabilistic acquisition model and use it to motivate our approach.
- We use novel smoothing and feature selection strategies motivated by the specificities of the hedge identification task.
- We analyse the presented techniques both theoretically and experimentally, reporting promising results for the task on a new publicly available full-text dataset.6

The work presented here has direct application in the biomedical research community. A key motivation for this study is to incorporate hedge identification into an interactive system for aiding curators in the construction and population of gene databases [17]. The system utilises sophisticated GUI (graphical user interface) and NLP technology to highlight key results within biomedical literature for expert curation. Our aim is to include a component that identifies when a result is presented in a speculative manner and should not be extracted. Recent research [23] suggests that approximately 30% of sentences in the results and discussion sections of biomedical papers contain speculative assertions, and this figure increases to around 40% for the conclusions section. Our analysis supports these estimates, and it is clear that interactive bioinformation systems that take account of hedging can render a significantly more effective service to curators and researchers alike.

We have presented our initial results on the hedge identification task in the hope that this will encourage others to carry the investigation further. Some potential avenues for future research are as follows:

- Active learning: Given a classifier trained on acquired data, a profitable subsequent step would be to apply active learning to further augment the training data with instances about which the classifier is uncertain. The combination of semi-supervised and active learning has been explored in various contexts, e.g. [20,25], and careful consideration would need to given to how best to combine the different models. It is our intuition that applying semi-supervised and active learning in series may be the best approach for our setting, rather than the more common method of combining them in parallel.
- Alternative semi-supervised learning strategies: It would be interesting to apply a completely different model type to the problem, for example label propagation as a variant of the semi-supervised paradigm. This would also facilitate the application of existing methods of combining semi-supervised and active learning in the graph based framework, for instance [37].

• **Representation:** There are various possibilities for taking account of context, e.g. which section of the paper a given sentence was drawn from, and whether its surrounding sentences are speculative. Explicit inclusion of negation might also be beneficial for improving recall of knowledge paucity hedges.

• **Acquisition phase stopping criteria:** An issue we have not addressed is that of whether the acquisition model can be automatically stopped at an optimal, or close to optimal point. Various methods have been investigated to address this problem, such as ‘counter-training’ [34] and committee agreement thresholding [36]; more work is needed to establish whether these or related ideas can be applied in our setting.

• **Assertion level hedge identification:** As mentioned earlier, rather than just knowing whether or not a sentence contains a hedge, it would be beneficial to know which assertion a given hedge scopes over. We propose that a sensible method would be to perform further analysis on the (relatively small) subset of sentences identified as belonging to the spec class to find the assertion boundaries and the scope of likely hedge cues. This would probably require a degree of syntactic analysis which could be derived from a dependency parser such as RASP.

**Acknowledgments**

This work was partially supported by the FlySlip project, BBSRC Grant BBS/B/16291, and we thank Ted Briscoe, Nikiforos Karamanis and Ruth Seal for thorough annotation and helpful discussion. The author is supported by a University of Cambridge Millennium Scholarship.

**References**


