A Generalised Hybrid Architecture for NLP

Alistair Willis
Department of Computing
The Open University,
Milton Keynes, UK
a.g.willis@open.ac.uk

Hui Yang
Department of Computing
The Open University,
Milton Keynes, UK
h.yang@open.ac.uk

Anne De Roeck
Department of Computing
The Open University,
Milton Keynes, UK
a.deroeck@open.ac.uk

Abstract

Many tasks in natural language processing require that sentences be classified from a set of discrete interpretations. In these cases, there appear to be great benefits in using hybrid systems which apply multiple analyses to the test cases. In this paper, we examine a general principle for building hybrid systems, based on combining the results of several, high precision heuristics. By generalising the results of systems for sentiment analysis and ambiguity recognition, we argue that if correctly combined, multiple techniques classify better than single techniques. More importantly, the combined techniques can be used in tasks where no single classification is appropriate.

1 Introduction

The success of hybrid NLP systems has demonstrated that complex linguistic phenomena and tasks can be successfully addressed using a combination of techniques. At the same time, it is clear from the NLP literature, that the performance of any specific technique is highly dependent on the characteristics of the data. Thus, a specific technique which performs well on one dataset might perform very differently on another, even on similar tasks, and even if the two datasets are taken from the same domain. Also, it is possible that the properties affecting the effectiveness of a particular technique may vary within a single document (De Roeck, 2007).

As a result of this, for many important NLP applications there is no single technique which is clearly to be preferred. For example, recent approaches to the task of anaphora resolution include syntactic analyses (Haghighi and Klein, 2009), Maximum Entropy models (Charniak and Elsner, 2009) and Support Vector Machines (Yang et al., 2006; Versley et al., 2008). The performance of each of these techniques varies depending upon the particular choice of training and test data.

This state of affairs provides a particular opportunity for hybrid system development. The overall performance of an NLP system depends on complex interactions between the various phenomena exhibited by the text under analysis, and the success of a given technique can be sensitive to the different properties of that text. In particular, the text's or document's properties are not generally known until the document comes to be analysed. Therefore, there is a need for systems which are able to *adapt* to different text styles at the point of analysis, and select the most appropriate combination of techniques for the individual cases. This should lead to hybridising techniques which are robust or adaptive in the face of varying textual styles and properties.

We present a generalisation of two hybridisation techniques first described in Yang et al. (2012) and Chantree et al. (2006). Each uses hybrid techniques in a detection task: the first is emotion detection from suicide notes, the second is detecting nocuous ambiguity in requirements documents. The distinguishing characteristic of both tasks is that a successful solution needs to accommodate uncertainty in the outcome. The generalised methodology described here is particularly suited to such tasks, where as well as selecting between possible solutions, there is a need to identify a class of instances where no single solution is most appropriate.

Proceedings of the Workshop on Innovative Hybrid Approaches to the Processing of Textual Data (Hybrid2012), EACL 2012, pages 97–105,
Avignon, France, April 23 2012.

2 Hybridisation as a Solution to Classification Tasks

The methodology described in this paper proposes hybrid systems as a solution to NLP tasks which attempt to determine an appropriate interpretation from a set of discrete alternatives, in particular where no one outcome is clearly preferable. One such task is nocuous ambiguity detection. For example, in sentence (1), the pronoun *he* could refer to *Bill*, *John* or to *John's father*.

(1) When Bill met John's father, he was pleased.

Here, there are three possible antecedents for *he*, and it does not follow that all human readers would agree on a common interpretation of the anaphor. For example, readers might divide between interpreting *he* as *Bill* or as *John's father*. Or perhaps a majority of readers feel that the sentence is sufficiently ambiguous that they cannot decide on the intended interpretation. These are cases of *nocuous ambiguity* (Chantree et al., 2006), where a group of readers do not interpret a piece of text in the same way, and may be unaware that the misunderstanding has even arisen.

Similarly, as a classification task, sentiment analysis for sentences or fragments may need to accommodate instances where multiple sentiments can be identified, or possibly none at all. Example (2) contains evidence of both *guilt* and *love*:

(2) Darling wife, — I'm sorry for everything.

Hybrid solutions are particularly suited to such tasks, in contrast to approaches which use a single technique to select between possible alternatives. The hybrid methodology proposed in this paper approaches such tasks in two stages:

1. Define and apply a set of heuristics, where each heuristic captures an aspect of the phenomenon and estimates the likelihood of a particular interpretation.

2. Apply a combination function to either combine or select between the values contributed by the individual heuristics to obtain better overall system performance.

The model makes certain assumptions about the design of heuristics. They can draw on a multitude of techniques such as a set of selection features based on domain knowledge, linguistic analysis and statistical models. Each heuristic is a *partial descriptor* of an aspect of a particular phenomenon and is intended as an "expert", whose opinion competes against the opinion offered by other heuristics. Heuristics may or may not be independent. The crucial aspect is that each of the heuristics should seek to *maximise precision* or complement the performance of another heuristic.

The purpose of step 2 is to maximise the contribution of each heuristic for optimal performance of the overall system. Experimental results analysed below show that selecting an appropriate mode of combination helps accommodate differences between datasets and can introduce additional robustness to the overall system. The experimental results also show that appropriate combination of the contribution of high precision heuristics significantly increases recall.

For the tasks under investigation here, it proves possible to select combination functions that allow the system to identify behaviour beyond classifying the subject text into a single category. Because the individual heuristics are *partial* descriptions of the whole language model of the text, it is possible to reason about the interaction of these partial descriptions, and identify cases where either none, or many, of the potential interpretations of the text are possible. The systems use either a machine learning technique or a voting strategies to combine the individual heuristics.

In sections 3 and 4, we explore how the previously proposed solutions can be classed as instances of the proposed hybridisation model.

3 Case study: Sentiment Analysis

Following Pang et al. (2002) and the release of the polarity 2.0 dataset, it is common for sentiment analysis tasks to attempt to classify text segments as either of positive or negative sentiment. The task has been extended to allow sentences to be annotated as displaying both positive and negative sentiment (Wilson et al., 2009) or indicating the degree of intensity (Thelwall et al., 2010).

The data set used for the 2011 i2b2 shared challenge (Pestian et al., 2012) differs from this model by containing a total of 15 different sentiments to classify the sentences. Each text fragment was labelled with zero, one or more of the 15 sentiments. For example, sentence (2) was annotated with both *Love* and *Guilt*. The fragments varied between phrases and full sentences, and the task aims to identify all the sentiments displayed by

each text fragment.

In fact, several of the proposed sentiments were identified using keyword recognition alone, so the hybrid framework was applied only to recognise the sentiments *Thankfulness*, *Love*, *Guilt*, *Hopelessness*, *Information* and *Instruction*; instances of the other sentiments were too sparse to be reliably classified with the hybrid system. A keyword cue list of 984 terms was manually constructed from the training data based on their frequency in the annotated set; no other public emotion lexicon was used. This cue list was used both to recognise the sparse sentiments, and as input to the CRF.

3.1 Architecture

An overview of the architecture is shown in figure 1. Heuristics are used which operate at the word level (Conditional Random Fields), and at the sentence level (Support Vector Machine, Naive Bayes and Maximum Entropy). These are combined using a voting strategy that selects the most appropriate combination of methods in each case.

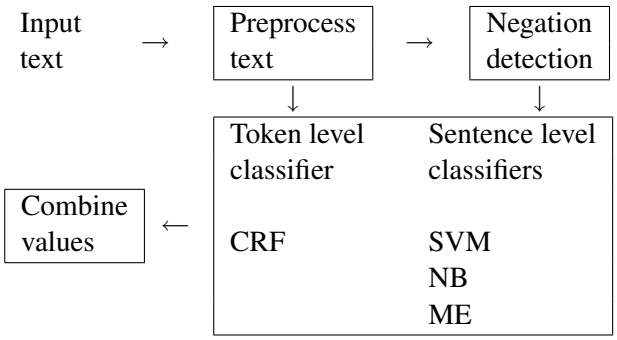

Figure 1: Architecture for sentiment classification task

The text is preprocessed using the tokeniser, POS tagger and chunker from the Genia tagger, and parsed using the Stanford dependency parser. This information, along with a negation recogniser, is used to generate training vectors for the heuristics. Negation is known to have a major effect on sentiment interpretation (Jia et al., 2009).

3.2 Sentiment recognition heuristics

The system uses a total of four classifiers for each of the emotions to be recognised. The only token-level classification was carried out using CRFs (Lafferty et al., 2001) which have been successfully used on Named Entity Recognition tasks. However, both token- and phrase-level recognition are necessary to capture cases where sentences convey more than one sentiment. The CRF-based classifiers were trained to recognise each of the main emotions based on the main keyword cues and the surrounding context. The CRF is trained on the set of features shown in figure 2, and implemented using CRF++[1].

Feature	Description
Words	word, lemma, POS tag, phrase chunk tag
Context	2 previous words and 2 following words with lemma, POS tags and chunk tags
Syntax	Dependency relation label and the lemma of the governer word in focus
Semantics	Is it negated?

Figure 2: Features used for CRF classifier

Three sentence-level classifiers were trained for each emotion, those being Naive Bayes and Maximum Entropy learners implemented by the MALLET toolkit[2], and a Support Vector Machine model implemented using SVM light[3] with the linear kernel. In each case, the learners were trained using a feature vector using the two feature vectors as shown in figure 3.

Feature vector	Description
Words	word lemmas
Semantics	negation terms identified by the negative term lexicon, and cue terms from the emotion term lexicon

Figure 3: Features used for sentence-level classifiers

A classifier was built for each of the main emotions under study. For each of the six emotions, four learners were trained to identify whether the text contains an instance of that emotion. That is, an instance of text receives 6 groups of results, and each group contains 4 results obtained from different classifiers estimating whether one particular emotion occurs. The combination function predicts the final sentiment(s) exhibited by the sentence.

[1] http://crfpp.sourceforge.net/
[2] http://mallet.cs.umass.edu/
[3] http://svmlight.joachims.org/

3.3 Combination function

To combine the outputs of the heuristics, Yang et al. (2012) use a voting model. Three different combination methods are investigated:

Any If a sentence is identified as an emotion instance by any one of the ML-based models, it is considered a true instance of that emotion.

Majority If a sentence is identified as an emotion instance by two or more of the ML-based models, it is considered a true instance of that emotion.

Combined If a sentence is identified as an emotion instance by two or more of the ML-based models *or* it is identified as an emotion instance by the ML-based model with the best precision for that emotion, it is considered a true instance of that emotion.

This combined measure reflects the intuition that where an individual heuristic is reliable for a particular phenomenon, then that heuristic's vote should be awarded a greater weight. The precision scores of the individual heuristics is shown in table 1, where the heuristic with the best precision for that emotion is highlighted.

Emotion	CRF	NB	ME	SVM
Thankfulness	**60.6**	58.8	57.6	52.6
Love	76.2	68.5	**77.6**	76.9
Guilt	58.1	46.8	35.3	**58.3**
Hopelessness	73.5	63.3	68.7	**74.5**
Information	53.1	41.0	48.1	**76.2**
Instruction	**76.3**	63.6	70.9	75.9

Table 1: Precision scores (%) for individual heuristics

3.4 Results

Table 2 reports the system performance on 6 emotions by both individual and combined heuristics.

In each case, the best performer among the four individual heuristics is highlighted. As can be seen from the table, the *Any* combinator and the *Combined* combinators both outperform each of the individual classifiers. This supports the hypothesis that hybrid systems work better overall.

3.5 Additional comments

The overall performance improvement obtained by combining the individual measures raises the question of how the individual elements interact. Table 3 shows the performance of the combined systems on the different emotion classes. For each emotion, the highest precision, recall and f-measure is highlighted.

As we would have expected, the *Any* strategy has the highest recall in all cases, while the *Majority* strategy, with the highest bar for acceptance, has the highest precision for most cases. The *Any* and *Combined* measures appear to be broadly comparable: for the measures we have used, it appears that the precision of the individual classifiers is sufficiently high that the combination process of improving recall does not impact excessively on the overall precision.

A further point of interest is that table 2 demonstrates that the Naive Bayes classifier often returns the highest f-score of the individual classifiers, even though it never has the best precision (table 1). This supports our thesis that a successful hybrid system can be built from multiple classifiers with high precision, rather than focussing on single classifiers which have the best individual performance (the *Combined* strategy favours the highest precision heuristic).

4 Nocuous ambiguity detection

It is a cornerstone of NLP that all text contains a high number of potentially ambiguous words or constructs. Only some of those will lead to misunderstandings, where two (or more) participants in a text-mediated interchange will interpret the text in different, and incompatible ways, without realising that this is the case. This is defined as nocuous ambiguity (Willis et al., 2008), in contrast to innocuous ambiguity, where the text is interpreted in the same way by different readers, even if that text supports different possible analyses.

The phenomenon of nocuous ambiguity is particularly problematic in high stake situations. For example, in software engineering, a failure to share a common interpretation of requirements stated in natural language may lead to incorrect system implementation and the attendant risk of system failure, or higher maintenance costs. The systems described by Chantree et al. (2006) and Yang et al. (2010a) aim not to *resolve* ambigu-

Emotion	Individual heuristics				Hybrid models		
	CRF	NB	ME	SVM	Any	Majority	Combined
Thankfulness	59.5	59.6	**61.9**	60.3	63.9	63.0	64.2
Love	63.7	**69.3**	66.5	61.5	72.0	70.3	71.0
Guilt	35.3	**40.5**	27.7	37.8	46.3	29.9	45.8
Hopelessness	63.2	**64.1**	59.9	57.0	67.3	65.4	67.3
Information	42.3	**47.7**	43.7	43.4	50.2	45.5	47.8
Instruction	**65.7**	65.7	63.4	58.8	72.1	65.4	72.0

Table 2: F-scores (%) for individual and combined heuristics (sentiment analysis)

	Any			Majority			Combined		
	P	R	F	P	R	F	P	R	F
Thankfulness	52.6	**81.6**	63.9	**60.6**	65.7	63.0	55.0	77.1	**64.2**
Love	68.7	**75.6**	**72.0**	**77.9**	64.0	70.3	74.6	67.7	71.0
Guilt	46.6	**46.2**	**46.3**	50.0	21.4	29.9	**50.5**	41.9	45.8
Hopelessness	64.1	**70.8**	67.3	**80.3**	55.2	65.4	66.3	68.4	**67.3**
Information	40.9	**64.9**	**50.2**	**49.9**	41.8	45.5	45.2	50.7	47.8
Instruction	68.5	**76.1**	**72.1**	**80.8**	54.9	65.4	70.3	73.7	72.0

Table 3: Precision, recall and F-scores (%) for the combined systems (sentiment analysis)

ous text in requirements, but to *identify* where instances of text might display nocuous ambiguity.

These systems demonstrate how, for hybrid systems, the correct choice of combination function is crucial to how the individual heuristics work together to optimise overall system performance.

4.1 Nocuous Ambiguity: Coordination

Chantree et al. (2006) focus on coordination attachment ambiguity, which occurs when a modifier can attach to one or more conjuncts of a coordinated phrase. For example, in sentence (3), readers may divide over whether the modifier *short* attaches to both *books* and *papers* (wide scope), or only to *books* (narrow scope).

(3) I read some short books and papers.

In each case, the coordination involves a near conjunct, (*books* in (3)), a far conjunct, (*papers*) and a modifier (*short*). The modifier might also be a PP, or an adverb in the case where a VP contains the conjunction. In disambiguation, the task would be to identify the correct scope of the modifier (i.e. which of two possible bracketings is the correct one). For nocuous ambiguity detection, the task is to identify to what extent people interpret the text in the same way, and to flag the instance as nocuous if they diverge relative to some threshold.

4.1.1 The dataset

17 human judgements were collected for each of 138 instances of sentences exhibiting coordination ambiguity drawn from a collection of software requirements documents. The majority of cases (118 instances) were noun compounds, with some adjective and some preposition modifiers (36 and 18 instances respectively). Participants were asked to choose between wide scope or narrow scope modifier attachment, or to indicate that they experienced the example as ambiguous. Each instance is assigned a *certainty* for wide and narrow scope modification reflecting the distribution of judgements. For instance, if 12 judges favoured wide scope for some instance, 3 judges favoured narrow scope and 1 judge thought the instance ambiguous, then the certainty for wide scope is 71% (12/17), and the certainty for narrow scope is 18% (3/17).

A key concept in nocuous ambiguity is that of an *ambiguity threshold*, τ. For some τ:

- if at least τ judges agree on the interpretation

of the text, then the ambiguity is *innocuous*,

- otherwise the ambiguity is *nocuous*.

So for $\tau = 70\%$, at least 70% of the judges must agree on an interpretation. Clearly, the higher τ is set, the more agreement is required, and the greater the number of examples which will be considered nocuous.

4.1.2 Selectional heuristics

A series of heuristics was developed, each capturing information that would lead to a preference for either wide or narrow scope modifier attachment. Examples from Chantree et al. (2006) propose seven heuristics, including the following:

Co-ordination Matching If the head words of the two conjuncts are frequently co-ordinated, this is taken to predict wide modifier scope.

Distributional Similarity If the head words of the two conjuncts have high distributional similarity (Lee, 1999), this is taken to predict wide modifier scope.

Collocation Frequency If the head word of the near conjunct has a higher collocation with the modifier than the far conjunct, this is taken to predict narrow modifier scope.

Morphology If the conjunct headwords have similar morphological markers, this is taken to predict wide modifier scope (Okumura and Muraki, 1994).

As with the sentiment recognition heuristics (section 3.2), each predicts one interpretation of the sentence with high precision, but potentially low recall. Recall of the *system* is improved by combining the heuristics, as described in the next section. Note that for the first three of these heuristics, Chantree et al. (2006) use the British National Corpus[4], accessed via the Sketch Engine (Kilgarriff et al., 2004), although a domain specific corpus could potentially be constructed.

4.1.3 Combining the heuristics

Chantree et al. (2006) combine the heuristics using the logistic regression algorithms contained in the WEKA machine learning package (Witten and Frank, 2005). The regression algorithm was trained against the training data so that the text was interpreted as nocuous *either* if there was evidence for both wide and narrow modifier scope *or* if there was no evidence for either.

This system performed reasonably for mid-range ambiguity thresholds (around $50\% < \tau < 80\%$; for high and low thresholds, naive baselines give very high accuracy). However, in subsequent work, Yang et al. (2010b) have demonstrated that by combining the results in a similar way, but using the LogitBoost algorithm, significant improvements can be gained over the logistic regression approach. Their paper suggests that LogitBoost provides an improvement in accuracy of up to 21% in the range of interest for τ over that of logistic regression.

We believe that this improvement reflects that LogitBoost handles interacting variables better than logistic regression, which assumes a linear relationship between individual variables. This supports our hybridisation method, which assumes that the individual heuristics can interact. In these cases, the heuristics bring into play different types of information (some structural, some distributional, some morphological) where each relies on partial information and favours one particular outcome over another. It would be unusual to find strong evidence of both wide and narrow scope modifier attachment from a single heuristic and the effect of one heuristic can modulate, or enhance the effect of another. This is supported by Chantree et al.'s (2006) observation that although some of the proposed heuristics (such as the morphology heuristic) perform poorly on their own, their inclusion in the regression model does improve the overall performance of the system

To conclude, comparing the results of Chantree et al. (2006) and Yang et al. (2010b) demonstrates that the technique of combining individual, high precision heuristics is a successful one. However, the combination function needs careful consideration, and can have as large an effect on the final results as the choice of the heuristics themselves.

4.2 Nocuous Ambiguity: Anaphora

As example (1) demonstrates, nocuous ambiguity can occur where there are multiple possible antecedents for an anaphor. Yang et al. (2010a) have addressed the task of nocuous ambiguity detection for anaphora in requirements documents, in sentences such as (4), where the pronoun *it* has

[4] http://www.natcorp.ox.ac.uk/

three potential antecedents (italicised).

(4) *The procedure* shall convert *the 24 bit image* to *an 8 bit image*, then display it in a dynamic window.

As with the coordination task, the aim is to identify nocuous ambiguity, rather than attempt to disambiguate the sentence.

4.2.1 The dataset

The data set used for the anaphora task consisted of 200 sentences collected from requirements documents which contained a third person pronoun and multiple possible antecedents. Each instance was judged by at least 13 people.

The concept of ambiguity threshold, τ, remains central to nocuous ambiguity for anaphora. The definition remains the same as in section 4.1.1, so that an anaphor displays innocuous ambiguity if there is an antecedent that at least τ judges agree on, and nocuous ambiguity otherwise. So if, say, 75% of the judges considered *an 8 bit image* to be the correct antecedent in (4), then the sentence would display nocuous ambiguity at $\tau = 80\%$, but innocuous ambiguity at $\tau = 70\%$.

For *innocuous* cases, the potential antecedent NP with certainty of at least τ is tagged as *Y*, and all other NPs are tagged as *N*. For *nocuous* cases, potential antecedents with τ greater than 0 are tagged as *Q* (questionable), or are tagged *N* otherwise ($\tau = 0$, ie. unselected).

4.2.2 Selectional Heuristics

The approach to this task uses only one selection function (Naive Bayes), but uses the output to support two different voting strategies. Twelve heuristics (described fully in Yang et al. (2010a)) fall broadly into three types which signal the likelihood that the NP is a possible antecedent:

linguistic such as whether the potential antecedent is a definite or indefinite NP

contextual such as the potential antecedent's recency, and

statistical such as collocation frequencies.

To treat a sentence, the classifier is applied to each of the potential antecedents and assigns a pair of values: the first is the predicted class of the antecedent (*Y*, *N* or *Q*), and the second is the associated probability of that classification.

Given a list of class assignments to potential antecedents with associated probabilities, a *weak positive threshold*, W_Y, and a *weak negative threshold*, W_N:

if the list of potential antecedents contains:
  one Y, no Q, one or more N
or
  no Y, one Q, one or more N but no *weak negatives*
or
  one *strong positive* Y , any number of Q or N
then
  the ambiguity is INNOCUOUS
else
  the ambiguity is NOCUOUS

where a classification Y is *strong positive* if its associated probability is greater than W_Y, and a classification N is *weak negative* if its associated probability is smaller than W_N.

Figure 4: Combination function for nocuous anaphora detection with weak thresholds

4.2.3 The combination function

As suggested previously, the choice of combination function can strongly affect the system performance, even on the same set of selectional heuristics. Yang et al. (2010a) demonstrate two different combination functions which exploit the selectional heuristics in different ways. Both combination functions use a voting strategy.

The first voting strategy states that a sentence exhibits innocuous ambiguity if either:

- there is a single antecedent labelled *Y*, and all others are labelled *N*, or
- there is a single antecedent labelled *Q*, and all others are labelled *N*.

The second strategy is more sophisticated, and depends on the use of *weak thresholds*: intuitively, the aim is to classify the text as innocuous if is (exactly) one *clearly* preferred antecedent among the alternatives. The combination function is shown in figure 4. The second clause states that a single potential antecedent labelled Q can be enough to suggest innocuous ambiguity if all the alternatives are N with a high probability.

	Model without weak thresholds			Model with weak thresholds		
τ	P	R	F	P	R	F
0.50	27.2	55.0	45.7	24.1	95.0	**59.7**
0.60	33.9	67.5	56.3	30.9	97.5	**68.1**
0.70	45.1	76.2	66.9	43.9	98.4	**78.8**
0.80	58.0	85.0	77.7	56.1	97.9	**85.5**
0.90	69.1	88.6	83.9	67.4	98.4	**90.1**
1.0	82.2	95.0	92.1	82.0	99.4	**95.3**

Table 4: Precision, Recall and f-measure (%) for the two combination functions (anaphora)

Task	Selectional heuristics	Combination functions
Sentiment analysis	CRF NB SVM ME	Voting - any - majority - combined
Nocuous ambiguity (coordination)	3 distributional metrics 4 others	logistic regression LogitBoost
Nocuous ambiguity (anaphora)	NB	Voting Voting (+ threshold)

Table 5: Hybridisation approaches used

The performance of the two voting strategies is shown in table 4. It is clear that the improved overall performance of the strategy with weak thresholds is due to the improved *recall* when the functions are combined; the precision is comparable in both cases. Again, this shows the desired combinatorial behaviour; a combination of high precision heuristics can yield good overall results.

5 Conclusion

The hybridised systems we have considered are summarised in table 5. This examination suggests that hybridisation can be a powerful technique for classifying linguistic phenomena. However, there is currently little guidance on principles regarding hybrid system design. The studies here show that there is room for more systematic study of the design principles underlying hybridisation, and for investigating systematic methodologies.

This small scale study suggests several principles. First, the sentiment analysis study has shown that a set of heuristics and a suitable combination function can outperform the best individually performing heuristic or technique. In particular, our results suggest that hybrid systems of the kind described here are most valuable when there is significant interaction between the various linguistic phenomena present in the text. This occurs both with nocuous ambiguity (where competition between the different interpretations creates disagreement overall), and with sentiment analysis (where a sentence can convey multiple emotions). As a result, hybridisation is particularly powerful where there are multiple competing factors, or where it is unclear whether there is sufficient evidence for a particular classification.

Second, successful hybrid systems can be built using multiple heuristics, even if each of the heuristics has low recall on its own. Our case studies show that with the correct choice of hybridisation functions, high precision heuristics can be combined to give good overall recall while maintaining acceptable overall precision.

Finally, the mode of combination matters. The voting system is successful in the sentiment analysis task, where different outcomes are not exclusive (the presence of *guilt* does not preclude the presence of *love*). On the other hand, the logitBoost combinator is appropriate when the different interpretations are exclusive (narrow modifier scope does preclude wide scope). Here, logitBoost can be interpreted as conveying the degree of uncertainty among the alternatives. The coordination ambiguity case demonstrates that the individual heuristics do not need to be independent, but if the method of combining them assumes independence, the benefits of hybridisation will be lost (logistic regression compared to LogitBoost).

This analysis has highlighted the interplay between task, heuristics and combinator. Currently, the nature of this interplay is not well understood, and we believe that there is scope for investigating the broader range of hybrid systems that might be applied to different tasks.

Acknowledgments

The authors would like to thank the UK Engineering and Physical Sciences Research Council who funded this work through the MaTREx project (EP/F068859/1), and the anonymous reviewers for helpful comments and suggestions.

References

Francis Chantree, Bashar Nuseibeh, Anne De Roeck, and Alistair Willis. 2006. Identifying nocuous ambiguities in natural language requirements. In *Proceedings of 14th IEEE International Requirements Engineering conference (RE'06)*, Minneapolis/St Paul, Minnesota, USA, September.

Eugene Charniak and Micha Elsner. 2009. EM works for pronoun anaphora resolution. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL '09)*, pages 148–156.

Anne De Roeck. 2007. The role of data in NLP: The case for dataset profiling. In Nicolas Nicolov, Ruslan Mitkov, and Galia Angelova, editors, *Recent Advances in Natural Language Processing IV*, volume 292 of *Current Issues in Linguistic Theory*, pages 259–266. John Benjamin Publishing Company, Amsterdam.

Aria Haghighi and Dan Klein. 2009. Simple coreference resolution with rich syntactic and semantic features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1152–1161, Singapore, August.

Lifeng Jia, Clement Yu, and Weiyi Meng. 2009. The effect of negation on sentiment analysis and retrieval effectiveness. In *The 18th ACM Conference on Information and Knowledge Management (CIKM'09)*, Hong Kong, China, November.

Adam Kilgarriff, Pavel Rychly, Pavel Smrz, and David Tugwell. 2004. The sketch engine. Technical Report ITRI-04-08, University of Brighton.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the International Conference on Machine Learning (ICML-2001)*, pages 282–289.

Lillian Lee. 1999. Measures of distributional similarity. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 25–32, College Park, Maryland, USA, June. Association for Computational Linguistics.

Akitoshi Okumura and Kazunori Muraki. 1994. Symmetric pattern matching analysis for english coordinate structures. In *Proceedings of the 4th Conference on Applied Natural Language Processing*, pages 41–46.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 79–86, Philadelphia, July.

John P. Pestian, Pawel Matykiewicz, Michelle Linn-Gust, Brett South, Ozlem Uzuner, Jan Wiebe, K. Bretonnel Cohen, John Hurdle, and Christopher Brew. 2012. Sentiment analysis of suicide notes: A shared task. *Biomedical Informatics Insights*, 5(Suppl 1):3–16.

Mike Thelwall, Kevan Buckley, Georgios Paltoglou, Di Cai, and Arvid Kappas. 2010. Sentiment in short strength detection informal text. *Journal of the American Society for Information Science & Technology*, 61(12):2544–2558, December.

Yannick Versley, Alessandro Moschitti, Massimo Poesio, and Xiaofeng Yang. 2008. Coreference systems based on kernels methods. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 961–968, Manchester, August.

Alistair Willis, Francis Chantree, and Anne DeRoeck. 2008. Automatic identification of nocuous ambiguity. *Research on Language and Computation*, 6(3-4):355–374, December.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2009. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational Linguistics*, 35(3):399–433.

Ian H. Witten and Eibe Frank. 2005. *Data mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2nd edition.

Xiaofeng Yang, Jian Su, and Chew Lim Tan. 2006. Kernel-based pronoun resolution with structured syntactic knowledge. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*, pages 41–48, Sydney, July.

Hui Yang, Anne De Roeck, Vincenzo Gervasi, Alistair Willis, and Bashar Nuseibeh. 2010a. Extending nocuous ambiguity analysis for anaphora in natural language requirements. In *18th International IEEE Requirements Engineering Conference (RE'10)*, Sydney, Australia, Oct.

Hui Yang, Anne De Roeck, Alistair Willis, and Bashar Nuseibeh. 2010b. A methodology for automatic identification of nocuous ambiguity. In *23rd International Conference on Computational Linguistics (COLING 2010)*, Beijing, China.

Hui Yang, Alistair Willis, Anne De Roeck, and Bashar Nuseibeh. 2012. A hybrid model for automatic emotion recognition in suicide notes. *Biomedical Informatics Insights*, 5(Suppl. 1):17–30, January.