Committed Belief Tagging on the FactBank and LU Corpora: A Comparative Study

Gregory J. Werner
Department of Computer Science
George Washington University
Washington, DC, USA
gwerner@gwu.edu

Vinodkumar Prabhakaran
Department of Computer Science
Columbia University
New York, NY, USA
vinod@cs.columbia.edu

Mona Diab
Department of Computer Science
George Washington University
Washington, DC, USA
mtdiab@gwu.edu

Owen Rambow
Center for Computational Learning Systems
Columbia University
New York, NY, USA
rambow@ccls.columbia.edu

Abstract

Level of committed belief is a modality in natural language, it expresses a speak-er/writers belief in a proposition. Initial work exploring this phenomenon in the literature both from a linguistic and computational modeling perspective shows that it is a challenging phenomenon to capture, yet of great interest to several downstream NLP applications. In this work, we focus on identifying relevant features to the task of determining the level of committed belief tagging in two corpora specifically annotated for the phenomenon: the LU corpus and the FactBank corpus. We perform a thorough analysis comparing tagging schemes, infrastructure machinery, feature sets, preprocessing schemes and data genres and their impact on performance in both corpora. Our best results are an F1 score of 75.7 on the FactBank corpus and 72.9 on the smaller LU corpus.

1 Introduction

Level of Committed belief (LCB) is a linguistic modality expressing a speaker or writer's (SW) level of commitment to a given proposition, which could be their own or a reported proposition. Modeling this type of knowledge explicitly is useful in determining an SWs cognitive state, also referred to as person's private state (Wiebe et al., 2005). Wiebe et al. (2005) use the definition of (Quirk et al., 1985), who defines a private state to be an "internal (state) that cannot be directly observed by others". Determining the cognitive state of an SW can be relevant to several natural language processing (NLP) tasks such as question answering, information extraction, confidence determination in people's deduced opinions, determining the veracity of information, understanding power/influence relations in linguistic communication, etc. As an example, in (Rosenthal and McKeown, 2012), LCB was used to improve their claim detector which in turn allowed for improvements in influence prediction.

Initial work addressed the task of automatically identifying LCB of the SW. Approaches to date have relied on supervised models dependent on manually annotated data. There are two standard annotated corpora, the LU corpus (Diab et al., 2009) and FactBank (Saurí and Pustejovsky, 2009). Though in effect aiming for the same objective, both corpora use different terminology, different annotation standards, and they cover different genres. Previous studies performed on these corpora were conducted independently. In this work, we explore both corpora systematically and investigate their respective proposed tag sets. We experiment with multiple machine learning algorithms, varying the tag sets as we go along. Our goal is to build an automatic LCB tagger that is robust in a multi-genre context. Eventually we aim to adapt this tagger to other languages. The LCB tagging task aims at automatically identifying beliefs which can be ascribed to a SW, and at identifying the strength level by which

he or she holds them. Across languages, many different linguistic devices are used to denote this attitude towards an uttered proposition, including syntax, lexicon, and morphology. In this work we focus our investigation of LCB tagging in English and we only address the problem from the perspective of the SW. We do not address nested LCB where the SW is reporting the LCB of other people (leading to nested attributions, as done in FactBank following the MPQA Sentiment corpus (Wiebe et al., 2005).

2 Background

Initial work on LCB was undertaken by Diab et al. (2009), who built the LU corpus that contains belief annotations for propositional heads in text. They used a 3-way distinction of belief tags: Committed Belief (CB) where the SW strongly believes in the proposition, Non-committed belief (NCB) where the SW reflects a weak belief in the proposition, and Non Attributable Belief (NA) where the SW is not (or could not be) expressing a belief in the proposition (e.g., desires, questions etc.). The LU corpus comprises over 13,000 word tokens from sixteen documents covering four genres: 9 newswire documents, 4 training manuals, 2 correspondences and 1 interview. One of the issues with this annotation scheme is that the annotations for NCB conflate the cases where the SW explicitly conveys the weakness of belief (e.g., using modal auxiliaries such as may) and the cases where the SW is reporting someone else's belief about a proposition. In this paper, we tease apart these original NCB cases and arrive at a 4-way belief distinction using the original annotations in the LU corpus (details to be discussed in Section 3.1).

The LCB tagger developed using the original LU corpus (Prabhakaran et al., 2010) obtained a best performance (64% F-measure) using the Yamcha[1] machine learning framework which leverages Support Vector Machines in a supervised manner, and a performance of 59% F-measure using the Conditional Random Fields (CRF) algorithm. Their experiments were limited in scope because the LU Corpus is fairly small. This led to an under-representation of NCB tags in the training corpus and a relatively shallow understanding of how LCB tagging performs across genres. In this paper, we perform a detailed investigation through extensive machine learning experiments to understand how the size of data and genre variations affect the performance of an LCB tagger. We also systematically measure the impact of augmenting the training data with more data as well as measuring performance differences when the training data comprises a single genre vs. multiple genres. It should be noted that although we experiment with similar machine learning frameworks, our results are not directly comparable since the Prabhakaran et al. (2010) work applied cross validation to the LU-3 corpus, while we did not follow the same experimental strategy. Additionally, in this work we use a lot more features than those reported in the previous study.

A closely related corpus is FactBank (FB; Saurí and Pustejovsky (2009)), which captures factuality annotations on top of event annotations in TimeML. FactBank is annotated on the genre of newswire. FactBank models the factuality of events at three levels: certain (CT), probable (PB) and possible (PS), and distinguishes the polarity (e.g., CT- means certainly not true). Moreover it marks an unknown category (Uu), which refers to uncommitted or underspecified belief. It also captures the source of the factuality assertions, thereby distinguishing the SW's factuality assertions from those of a source introduced by the author. Despite the terminology difference between FactBank ("factuality") and LU ("committed belief"), they both address the same type of linguistic modality phenomenon namely level of committed belief. Accordingly, with the appropriate mapping, both corpora can be used in conjunction to model LCB. From a computational perspective, FactBank differs from the LU corpus in two major respects (other than the granularity in which they capture annotations): 1) FactBank is roughly four times the size of the LU corpus, and 2) FactBank is more homogeneous in terms of genre than the LU corpus as it consists primarily of newswire. In this paper, we unify the factuality annotations in Factback and the level of committed belief annotations present in the LU corpus to a 4-way committed-belief distinction.[2]

[1] http://chasen.org/∼taku/software/yamcha/

[2] For an additional discussion of the relation between factuality and belief, see (Prabhakaran et al., 2015)

3 Approach

Following previous work (Prabhakaran et al., 2010), we adopt a supervised approach to the LCB problem. We experiment with the two available manually annotated corpora, the LU and FB corpora. Going beyond previous approaches to the problem reported in the literature, our goal is to create a robust LCB system while gaining a deeper understanding of the phenomenon of LCB as an expressed modality by systematically teasing apart the different factors that affect performance.

3.1 Annotation Transformations

The NCB category of the LU tagging scheme captures two different notions: that of uncertainty of the speaker/writer and that of belief being attributed to someone other than the SW. Accordingly, we manually split the NCB into the NCB tag and the Reported Belief tag (ROB). Reported belief is the case where the SW's intention is to report on someone else's stated belief, whether or not they themselves believe it. An example of this would be the sentence *John said he studies everyday*. While the 'say' proposition is an example of committed belief (CB) on the part of the SW, the SW makes no assertion about the 'study' proposition attributed to John, and therefore *studies* is labeled ROB. This relabeling of the NCB tag into NCB and ROB was carried out manually by co-authors Werner and Rambow, who are native speakers of English. The inter-annotator agreement was 93%. The cases where there were contentions were discussed and an adjudication process was followed where a single annotation was agreed upon. This was a relatively fast process since the number of NCB annotated data is very small in the original LU corpus (176 instances). This conversion resulted in the LU-4 corpus designating the fact that this version of the LU corpus is a 4-way annotated corpus. This is in contrast to the original version of LU corpus with the 3-way distinction, LU-3.

To illustrate the difference between each of the tags in the LU-4 corpus, we provide a few examples from the annotated corpus. The sentence 1 shows the contrast between the committed belief in the author knowing and the non-committed belief in the author being uncertain of it (a flu vaccine) working. The other two tags are demonstrated in the sentence 2 where the author is saying Reed accused however Reed is the one talking about failing and not the author. To contrast we note that although Reed is attributed the notion of failing, neither the author nor Reed demonstrate any belief of the verb to probe and therefore it is not-attributable to any source mentioned in this sentence.

(1) But we only <CB> know </CB> that it might <NCB> work </NCB> because of laboratory studies and animal studies uh uh

(2) Democratic leader Harry Reed <CB> accused </CB> Republicans of <ROB> failing </ROB> to <NA> probe </NA> allegations ...

In order to render the FactBank (FB) corpus comparable to the LU-4 corpus, we mapped tags in the FB corpus into the 4-way tag scheme adopted in the LU-4 framework. Accordingly, we mapped CT directly into CB, PB and PS directly into NCB, and Uu was mapped into either NA or ROB. We used the number and identity of sources to determine if the Uu of FB was due to belief expressed by a source other than the SW. Specifically, if the same proposition is marked Uu for the SW, but the annotations also capture factuality attributed to another source, then we conclude the tag should be ROB. If there is no other attribution on the proposition other than the Uu attributed to the SW, we consider the tag to be NA. We refer to the resulting version of the FB corpus as FB-4. It is worth noting that because the genre of the FB corpus is newswire, it has a relatively large number of ROB annotations. Moreover, FB explicitly marks LCB with respect to various nested sources. However in our mapping, we only consider the annotations from the perspective of the SW.

We give a few examples of the original FactBank work as to compare and contrast the notion of belief carried in each corpus. In sentence 3, we have clear cut mapping between the certainty of the author in think and committed belief. Likewise, doing is a non-committed belief. In each case, the polarity is discarded in our transformations. The sentence 4 reveals the case where teaches takes on a reported belief meaning as it is given both a certain tag for the school and an unknown tag for the author. An example where Uu does not constitute reported belief is shown in the sentence 5, where only one entity's

belief is conveyed, and that is of the author.

(3) ... Yeah I <CT+> think </CT+> he's <PR+> doing </PR> the right thing

(4) The school <CT+> says </CT+> it <CT+> <Uu> teaches </Uu> </CT+> the children to be good Muslims and good students.

(5) I <CT+> urge <CT+> you to <Uu> do <Uu> the right thing ...

The tag distribution breakdown in the corpora is illustrated in Table 1.

	CB	NCB	ROB	NA	Total
LU-3	631	176		589	13485
LU-4	631	15	161	589	13485
FB-4	3837	156	2074	661	82845

Table 1: Label distribution in the LU-3, LU-4 and FB-4 corpora.

3.2 Experimental Set Up

3.2.1 Corpus Combination

We experiment with the three corpora LU-3 (with the labels CB, NCB and NA), LU-4 and FB-4 (each with the labels CB, NCB, NA, and ROB). We present results on each of the corpora and their combinations for training and testing. In general we split our corpora at the sentence level into training and test sets with 5/6 for training and 1/6 for test by reserving every sixth sentence for the test set.

3.2.2 Features

We use a number of features proposed by Prabhakaran et al. (2010), as well as a few more recent additions, and we hold them constant across our different experimental conditions. This feature set comprises the following base set of lexical and syntactic based features. *General Features* for each token include its lemma, part of speech (POS), as well as the lemma and POS of two preceding and following tokens. *Dependency features* include sibling's lemma, sibling's POS, child's lemma, child's POS, parent's lemma, parent's POS, ancestors' lemmas, ancestor's POS, reporting ancestor's POS, reporting ancestor's lemma, dependency relationship and lemma of the closest ancestor whose POS is a noun, dependency relationship and lemma of the closest ancestor whose POS is a verb, token under the scope of a conditional (*if/when*), ancestor under the scope of a conditional (*if/when*). We use the Dependency Parser provided in the Stanford NLP toolkit. A pictorial explanation of some of these dependency features is given in Figure 1 and Table 2 for the sentence "Republican leader Bill Frist said the Senate was hijacked".

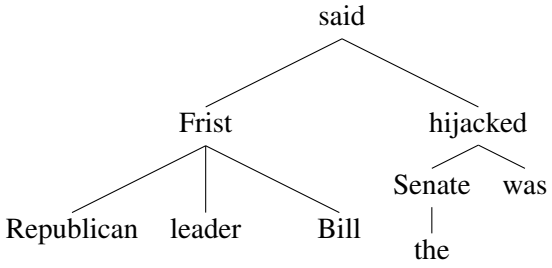

Figure 1: Dependency tree for example sentence.

Feature Name	Value
PosTag	VBN
Lemma	hijack
WhichModalAmI	nil
UnderConditional	N/A
AncestorUnderConditional	N/A
FirstDepAncestorofPos	{hijack, NIL}, {say, ccomp}
DepAncestors	{say,VBD}
Siblings	{Frist, NNP}
Parent	{say, VBD, say-37.7-1}
Child	{Senate, NNP}, {be, VBD}
DepRel	{ccomp}

Table 2: Representative features for the token hijacked in the example sentence.

3.2.3 Machine Learning Infrastructure

We experiment with five machine learning algorithms. A. Conditional Random Fields (CRF) to allow for comparison with previous work; B. Linear Support Vector Machines (LSVM); C. Quadratic Kernel Support Vector Machines (QSVM); D. Naive Bayes (NB); and, E. Logistic Regression (LREG). We provide NB as a generative contrast to the discriminative SVM and CRF methods. Moreover,

QSVM quite often yields better results at the expense of longer runtime, hence, we explore if that is the case within the LCB task.

The parameters for each of the five algorithms are held constant across all experiments and not tuned for specific configurations. The notable parameters that are used are listed in Table 3.

Algorithm	Notable Parameters
CRF	Gaussian Variance=10, Orders = 1, Iterations = 500
LSVM	Linear Kernel (t=0), Classification (z=c), Cost Factor (j=1), Biased Hyperplane (b=1), Do not remove inconsistent training examples (i=0)
QSVM	Polynomial Kernel (t=1), Quadratic (d=1), Classification (z=c), Cost Factor (j=1), Biased Hyperplane (b=1), Do not remove inconsistent training examples (i=0)
NB	Default
LREG	Default

Table 3: Parameter settings per algorithm.

3.2.4 Tools

A list of major NLP tools used is illustrated in Table 4. We used the CoreNLP pipeline for tokenization, sentence splitting, part of speech determination, lemmatization, named entity recognition, dependency parsing and coreference resolution. ClearTK provided us easy access to the machine learning algorithms we used which includes SVM Light for both SVM kernels and Mallet for CRF. It also provides us the backbone for our annotation structure.

4 Experiments

4.1 Evaluation metric

We ran 30 experiments, which are all the possible permutations of the three variables, listed above: do we split the NCB tag into 2 tags, what corpora do we train on, and what machine learning algorithm do we use. We report results using the overall weighted micro average F1 score.

Name	Source	Ver.
CoreNLP	(Manning et al., 2014)	3.5
ClearTK	(Bethard et al., 2014)	2.0
UIMA	https://uima.apache.org/	2.6
uimaFIT	(Ogren and Bethard, 2009)	2.1
Mallet	(McCallum, 2002)	2.0.7
SVM Light	(Joachims, 1999)	6.0.2

Table 4: NLP Tools Used.

4.2 Condition 1: Impact of splitting NCB tag in the LU corpus

We show the overall impact of splitting the NCB tag in the original LU corpus into two tags: NCB and ROB. The training and test corpora are from the same corpus, i.e. training and test sets are from LU-3 or LU-4. The results are reported on the respective test sets using the F1 score. The hypothesis is that a 4-way tagging scheme should result in better overall scores if the tagging scheme indeed captures a more genuine explicit representation for LCB. Table 5 illustrates the results yielded from the 5 ML algorithms. We note that the 4-way tagging outperforms the 3-way tagging for CRF and LSVM, however, the NB algorithm doesn't seem as sensitive to the tagging scheme (3 vs. 4 tags), and QSVM and LREG seem to be better performing in the 3 tag setting than the 4 tag setting. This might be a result of the number of tags in the 4-way tagging scheme breaking up the space for NCB's considerably. Overall the highest score is obtained by LSVM (72.89 F1 score) for LU-4, namely in the 4-way tagging scheme, suggesting that a 4-way split of the annotation space is an appropriate level of annotation granularity.

4.3 Condition 2: Impact of size and corpus genre homogeneity on LCB performance

In this condition we attempt to tease apart the impact of corpus size (FB being 4 times the size of the LU corpus) as well as corpus homogeneity, since FB is relatively homogeneous in genre compared to the LU corpus. Similar to Condition 1, we show the results yielded by all 5 ML algorithms. Results are reported in Table 6. Our hypothesis is that the overall results obtained on the FB should outper-

Test Set	Algorithm	Overall F-score
LU-4	CRF	71.33
LU-4	**LSVM**	**72.89**
LU-4	QSVM	68.10
LU-4	NB	61.61
LU-4	LREG	70.75
LU-3	CRF	68.25
LU-3	LSVM	69.77
LU-3	QSVM	69.21
LU-3	NB	61.58
LU-3	**LREG**	**71.26**

Table 5: Condition 1: LU-3 and LU-4 results using micro average F1 score on their respective test data.

Test Set	Algorithm	Overall F-score
FB-4	CRF	73.34
FB-4	LSVM	74.36
FB-4	**QSVM**	**75.57**
FB-4	NB	66.22
FB-4	LREG	74.67
LU-4	CRF	71.33
LU-4	**LSVM**	**72.89**
LU-4	QSVM	68.10
LU-4	NB	61.61
LU-4	LREG	70.75

Table 6: Condition 2: FB-4 and LU-4 results using micro average F1 score on their respective test data.

form those obtained on the LU corpus. Note that the results in the Table 6 are not directly comparable across corpora since the test sets are different: each experimental condition is tested within the same corpus, i.e. FB-4 is trained using FB-4 training data and tested on FB-4 test data, and LU-4 is trained using LU-4 training data and tested on LU-4 test data. However, the results validate our hypothesis that more data which is more homogeneous results in a better LCB tagger.

It is noted that the various ML algorithms perform differently for LU-4 vs. FB-4. In order, for FB-4, QSVM outperforms LREG which in turn outperforms LSVM, CRF and NB. In contrast, for LU the LSVM is the best performing ML algorithm followed by CRF, QSVM, LREG, and finally NB. The linear kernel SVM, LSVM, has the closest performance between the two, yet the difference is still statistically significant.

A deeper analysis on each of the four tags shows a remarkable difference in F1-measure for reported belief (ROB) for the two corpora as illustrated in Table 7. ROB is significantly better identified in the FB-4 corpus compared to the LU-4 corpus. This is expected since the FB-4 corpus has significantly more ROB tags in the training data. The number of ROB tags in training sets for LU-4 is 100 and for FB-4 it is 1800. The NA tag on the other hand performs better in the LU-4 corpus than in the FB-4 as seen in Table 8. The number of NA tags in the LU-4 training data is 460, while in the FB-4 training data (which is much larger) there are 600. In the case of FB-4 they only constitute a small percentage of the overall data compared to their percentage in the LU-4 corpus.

Test Set	Algorithm	P	R	F
LU-4	**CRF**	**66.67**	**40.00**	**50.00**
LU-4	LSVM	39.13	45.00	41.86
LU-4	QSVM	50.00	15.00	23.08
LU-4	NB	0.00	0.00	0.00
LU-4	LREG	41.67	25.00	31.25
FB-4	CRF	76.79	73.22	74.97
FB-4	LSVM	76.08	72.13	74.05
FB-4	**QSVM**	**72.86**	**79.23**	**75.92**
FB-4	NB	57.27	67.76	62.08
FB-4	LREG	74.59	73.77	74.18

Table 7: ROB results in FB-4 and LU-4 Corpora.

Test Set	Algorithm	P	R	F
LU-4	CRF	70.59	77.42	73.85
LU-4	**LSVM**	**80.21**	**82.80**	**81.48**
LU-4	QSM	64.91	79.57	71.50
LU-4	NB	66.32	67.74	67.02
LU-4	LREG	76.00	81.72	78.76
FB-4	CRF	52.38	44.72	48.25
FB-4	LSVM	50.74	56.10	53.28
FB-4	**QSVM**	**61.76**	**51.22**	**56.00**
FB-4	NB	0.00	0.00	0.00
FB-4	LREG	54.63	47.97	51.08

Table 8: NA results in FB-4 and LU-4 corpora.

4.4 Condition 3: Measuring impact of training data size on performance: combining training FB-4 and LU-4 data

In this condition, we wanted to investigate the impact of training using the combined FB-4 and LU-4 training corpora on 3 test sets: LU-4 Test, FB-4 Test and LU-4 Test + FB-4 Test. A reasonable hypothesis is that, with a larger corpus created by combining the two individual corpora we will see better results on any test corpus. Table 9 illustrates the experimental results for this condition where the training data for both corpora are combined.

The worst overall results are obtained on the LU-4 test set, while the best are obtained on the FB-4 test set. This is expected since the size of the training data coming from the FB-4 corpus overwhelms that of the LU corpus and the LU corpus is relatively diverse in genre, potentially adding noise. Also we note that the results on the LU-4 corpus are much worse than the results obtained and illustrated in Table 5 when the training data was significantly smaller, yet of strictly the same genre of the test data. This observation seems to suggest that homogeneity between training and test data for the LCB task trumps training data size. We also note that this observation is furthermore supported by the slight degradation in performance in the FB-4 test set compared to the performance results reported in Table 6 for the ML algorithms CRF and QSVM. However, we observe that LREG, NB and LSVM each was

Test Set	Algorithm	Overall F-score
LU-4	CRF	56.30
LU-4	**LSVM**	**61.10**
LU-4	QSVM	58.05
LU-4	NB	45.32
LU-4	LREG	59.90
FB-4	CRF	73.02
FB-4	LSVM	74.99
FB-4	QSVM	75.00
FB-4	NB	67.37
FB-4	**LREG**	**75.23**
FB-4 + LU-4	CRF	70.47
FB-4 + LU-4	LSVM	72.85
FB-4 + LU-4	QSVM	72.48
FB-4 + LU-4	NB	64.02
FB-4 + LU-4	**LREG**	**72.88**

Table 9: Condition 3: Micro average F1 score results obtained on three sets of test data while trained using a combination of FB-4 and LU-4 training data.

able to generalize better from the augmented data when additionally using the LU-4 training data, but the improvements were relatively insignificant (less than 1%). This may be attributed to the addition of the LU-4 training data, which adds noise to the LCB training task leading to inconclusive results. Testing on a combined corpus shows that LREG algorithm yields the best results.

4.5 Condition 4: Machine Learning Algorithm Performance

From the first three conditions, we are able to conclude how reliably certain machine learning algorithms outperform others. In our research, we have mainly focused on SVM Light's linear kernel (LSVM) and expect it to perform quite well. Certainly, we would expect it to outperform the CRFs, as they did in previous work. Changing the linear kernel to a quadratic kernel might give us some improvement at the expense of training time since it takes much longer to complete. Our intuition as far as CRFs being outperformed by SVMs seem to hold

uniformly as Tables 5, 6 and 9 illustrate. To augment the linear kernel SVM, the quadratic kernel only gives an improvement in some cases.

The NB models performed predictably poorly. Surprisingly, the LREG models appear to be robust, with performance that is comparable to the best SVM models (LSVM and QSVM) in our experiments. In fact, for the FB-4 case, LREG performed slightly better than either LSVM or QSVM. Given the efficiency of LREG in terms of training and testing, and its comparable performance to SVMs, using LREG for feature exploration in the context of LCB tagging makes it a very attractive ML framework to tune parameters with, keeping the more sophisticated ML algorithms for final testing.

Sometimes it is other components that cause an error. Take this example sentence from the LSVM algorithm acting on the LU corpus: It also checks on guard posts. Checks has been annotated CB and correctly so by the human involved. The tagger marks checks as O, or lacking author belief, because the token checks has been labeled a noun by the part-of-speech checker. A more proper miss can be found on the sentence You know what's sort of interesting Paula once again taken from the LU corpus. Although labeled as NA, the token know is labeled as CB. Since it has the feel of a question the annotator has stated that there is no committed belief on the part of the author. This is one that the algorithm itself has clearly gotten wrong. The CRF on the same sentence chose O, or lack of belief. NB got the token correct. LREG chose O. QSVM took the same approach as the LSVM labeling it as CB. This illustration shows the worse performing algorithm on the LU-4 corpus being the only correct answer showing perhaps that detecting phrases and sentences formed as questions are harder to analyze.

5 Conclusions

The results suggest that 4-way LCB tagging is an appropriate LCB granularity level. Training and testing on the FB-4 corpus results in overall better performance than training and testing on the LU corpus. We have seen that the LCB task is quite sensitive to the consistency in genre across training and test data, and that more out-of-genre data is not always the best route to overall performance improvement. SVMs were one of the best performing ML platforms in the context of this task as well as Logistic regression.

Acknowledgements

This paper is based upon work supported by the DARPA DEFT Program. The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government. We also thank the anonymous reviewers for their constructive feedback.

References

Steven Bethard, Philip Ogren, and Lee Becker. 2014. ClearTK 2.0: Design Patterns for Machine Learning in UIMA. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3289–3293, Reykjavik, Iceland, 5. European Language Resources Association (ELRA).

Mona Diab, Lori Levin, Teruko Mitamura, Owen Rambow, Vinodkumar Prabhakaran, and Weiwei Guo. 2009. Committed Belief Annotation and Tagging. In *Proceedings of the Third Linguistic Annotation Workshop*, pages 68–73, Suntec, Singapore, August.

Thorsten Joachims. 1999. Making Large-Scale SVM Learning Practical. In Bernhard Schölkopf, Christopher J.C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, Cambridge, MA, USA. MIT Press.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland, June.

Andrew Kachites McCallum. 2002. MALLET: A Machine Learning for Language Toolkit. http://www.cs.umass.edu/ mccallum/mallet.

Philip Ogren and Steven Bethard. 2009. Building Test Suites for UIMA Components. In *Proceedings of the Workshop on Software Engineering, Testing, and Quality Assurance for Natural Language Processing (SETQA-NLP 2009)*, pages 1–4, Boulder, Colorado, June.

Vinodkumar Prabhakaran, Owen Rambow, and Mona Diab. 2010. Automatic Committed Belief Tagging. In *Coling 2010: Posters*, pages 1014–1022, Beijing, China, August. Coling 2010 Organizing Committee.

Vinodkumar Prabhakaran, Tomas By, Julia Hirschberg, Owen Rambow, Samira Shaikh, Tomek Strzalkowski,

Jennifer Tracey, Michael Arrigo, Rupayan Basu, Micah Clark, Adam Dalton, Mona Diab, Louise Guthrie, Anna Prokofieva, Stephanie Strassel, Gregory Werner, Janyce Wiebe, and Yorick Wilks. 2015. A New Dataset and Evaluation for Belief/Factuality. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics (*SEM 2015)*, Denver, USA, June.

Randolph Quirk, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. 1985. *A Comprehensive Grammar of the English Language*. Longman, London.

Sara Rosenthal and Kathleen McKeown. 2012. Detecting Opinionated Claims in Online Discussions. In *Semantic Computing (ICSC), 2012 IEEE Sixth International Conference on*, pages 30–37. IEEE.

Roser Saurí and James Pustejovsky. 2009. FactBank: A Corpus Annotated with Event Factuality. *Language Resources and Evaluation*, 43:227–268. 10.1007/s10579-009-9089-9.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating Expressions of Opinions and Emotions in Language. *Language Resources and Evaluation*, 39(2-3):165–210.