Conditional Random Fields for Identifying Appropriate Types of Support for Propositions in Online User Comments

Joonsuk Park
Dept. of Computer Science
Cornell University
Ithaca, New York, USA
jpark@cs.cornell.edu

Arzoo Katiyar
Dept. of Computer Science
Cornell University
Ithaca, New York, USA
arzoo@cs.cornell.edu

Bishan Yang
Dept. of Computer Science
Cornell University
Ithaca, New York, USA
bishan@cs.cornell.edu

Abstract

Park and Cardie (2014) proposed a novel task of automatically identifying appropriate types of support for propositions comprising online user comments, as an essential step toward automated analysis of the adequacy of supporting information. While multiclass Support Vector Machines (SVMs) proved to work reasonably well, they do not exploit the sequential nature of the problem: For instance, verifiable experiential propositions tend to appear together, because a personal narrative typically spans multiple propositions. According to our experiments, however, Conditional Random Fields (CRFs) degrade the overall performance, and we discuss potential fixes to this problem. Nonetheless, we observe that the F_1 score with respect to the unverifiable proposition class is increased. Also, semi-supervised CRFs with posterior regularization trained on 75% labeled training data can closely match the performance of a supervised CRF trained on the same training data with the remaining 25% labeled as well.

1 Introduction

The primary domain for argumentation mining has been professionally written text, such as parliamentary records, legal documents and news articles, which contain well-formed arguments consisting of explicitly stated premises and conclusions (Palau and Moens, 2009; Wyner et al., 2010; Feng and Hirst, 2011; Ashley and Walker, 2013). In contrast, online user comments are often comprised of *implicit* arguments, which are conclusions with no explicitly stated premises[1]. For instance, in the following user comment, neither of the two propositions are supported with a reason or evidence. In other words, each of the two propositions is the conclusion of its own argument, with no explicit support provided (thus called *implicit* arguments):

> All airfare costs should include the passenger's right to check at least one standard piece of baggage.$_A$ All fees should be fully disclosed at the time of airfare purchase, regardless of nature.$_B$

When the goal is to extract well-formed arguments from a given text, one may simply disregard such implicit arguments. (Villalba and Saint-Dizier, 2012; Cabrio and Villata, 2012). However, with the accumulation of a large amount of text consisting of implicit arguments, a means of assessing the adequacy of support in arguments has become increasingly desirable. It is not only beneficial for analyzing the strength of arguments, but also for helping commenters to construct better arguments by suggesting the appropriate types of support to be provided.

As an initial step toward automatically assessing the adequacy of support in arguments, Park and Cardie (2014) proposed a novel task of classifying each proposition based on the appropriate type of support: unverifiable (UNVERIF), verifiable non-experiential (VERIF$_{NON}$), or verifiable experiential

[1]Note that implicit arguments are different from so called *enthymemes*, which may contain explicit premises, along with one or more missing premises.

Proceedings of the 2nd Workshop on Argumentation Mining, pages 39–44,
Denver, Colorado, June 4, 2015.

(VERIF$_{EXP}$)[2]. They show that multiclass Support Vector Machines (SVMs) can perform reasonably well on this task.

SVMs, however, do not leverage on the sequential nature of the propositions: For instance, when a commenter writes about his past experience, it typically spans multiple propositions. (In our dataset, VERIF$_{EXP}$ is followed by VERIF$_{EXP}$ with 57% probability, when VERIF$_{EXP}$ constitutes less than 15% of the entire dataset.) Thus, we expect that the probability of a proposition being a verifiable experiential proposition significantly increases when the previous proposition is a verifiable experiential proposition.

In this paper, we test our intuition by employing Conditional Random Field (CRF), a popular approach for building probabilistic models to classify sequence data, for this task (Lafferty et al., 2001). In addition, we experiment with various ways to train CRFs in a semi-supervised fashion.

Unlike our intuition, we find that a CRF performs worse than a multiclass SVM overall. Still, the F_1 score with respect to the UNVERIF class is improved. Also, we show that semi-supervised CRFs with posterior regularization trained on 75% labeled training data can closely match the performance of a supervised CRF trained on the same training data with the remaining 25% labeled as well.

2 Appropriate Support Type Identification

2.1 Task

The task is to classify a given proposition based on the type of appropriate support. In this subsection, we give a brief overview of the target classes[3].

Verifiable Non-experiential (VERIF$_{NON}$). Propositions are verifiable if its validity can be proved/disproved with objective evidence. Thus, it cannot contain subjective expressions, and there should be no room for multiple subjective interpretations. Also, assertions about the future is considered unverifiable, as its truthfulness cannot be confirmed at the present time. As the propositions of this type are verifiable, the appropriate type of support is objective evidence. ("Non-experiential" here means that the given proposition is not about a personal state or experience. The reason for making this distinction is discussed in the next paragraph.)

Verifiable Experiential (VERIF$_{EXP}$). The only difference between this class and VERIF$_{NON}$ is that this type of propositions is about a personal state or experience. Verifiable propositions about a personal state or experience are unique in that it can be inappropriate to evidence for them: People often do not have objective evidence to prove their past experiences, and even if they do, providing it may violate privacy. Thus, the appropriate type of support for this class is still evidence, but optional.

Unverifiable (UNVERIF). Propositions are unverifiable if they contain subjective opinions or judgments, as the subjective nature prevents the propositions from having a single truth value that can be proved or disproved with objecive evidence. Also, assertions about a future event is also unverifiable, because the future has not come yet. As there is no objective evidence for this type of propositions, the appropriate type of support is a *reason*.

Other Statement (OTHER). The remainder of user comments, i.e. text spans that are not part of an argument, falls under this category. Typical examples include questions, greetings, citations and URLs. Among these, only citations and URLs are considered argumentative, as they can be used to provide objective evidence. Luckily they can be accurately identified with regular expressions and thus are excluded from his classification task.

2.2 Conditional Random Fields

We formulate the classification task as a sequence labeling problem. Each user comment consists of a sequence of propositions (in the form of sentences or clauses), and each proposition is classified based on its appropriate support type. Instead of predicting the labels individually, we jointly optimize for the sequence of labels for each comment.

We apply CRFs (Lafferty et al., 2001) to the task as they can capture the sequence patterns of propositions. Denote $\mathbf{x}$ as a sequence of propositions within a user comment and $\mathbf{y}$ as a vector of labels. The CRF

[2]See Section 2 for a more information.

[3]For more details with examples, please refer to the original paper.

models the following conditional probabilities:

$$p_\theta(\mathbf{y}|\mathbf{x}) = \frac{\exp(\theta \cdot f(\mathbf{x}, \mathbf{y}))}{Z_\theta(\mathbf{x})}$$

where $f(\mathbf{x}, \mathbf{y})$ are the model features, θ are the model parameters, and $Z_\theta(\mathbf{x}) = \sum_{\mathbf{y}} \exp(\theta \cdot f(\mathbf{x}, \mathbf{y}))$ is a normalization constant. The objective function for a standard CRF is to maximize the log-likelihood over a collection of labeled documents plus a regularization term:

$$\max_\theta \mathcal{L}(\theta) = \max_\theta \sum_{(\mathbf{x},\mathbf{y})} \log p_\theta(\mathbf{y}|\mathbf{x}) - \frac{||\theta||_2^2}{2\delta^2}$$

Typically CRFs are trained in a supervised fashion. However, as labeled data is very difficult to obtain for the task of support identification, it is important to exploit distant supervision in the data to assist learning. Therefore, we investigate semi-supervised CRFs which train on both labeled and unlabeled data by using the posterior regularization (PR) framework (Ganchev et al., 2010). PR has been successfully applied to many structured NLP tasks such as dependency parsing, information extraction and sentiment analysis tasks (Ganchev et al., 2009; Bellare et al., 2009; Yang and Cardie, 2014).

The training objective for semi-supervised CRFs augments the standard CRF objective with a posterior regularizer:

$$\begin{aligned} \max_\theta \mathcal{L}(\theta) - \min_{q \in \mathcal{Q}} \{ & KL(q(\mathbf{Y})||p_\theta(\mathbf{Y}|\mathbf{X})) \\ & + \beta||E_q[\phi(\mathbf{X}, \mathbf{Y})] - \mathbf{b}||_2^2\} \end{aligned} \quad (1)$$

The idea is to find an optimal auxiliary distribution q that is closed to the model distribution $p_\theta(\mathbf{Y}|\mathbf{X})$ (measured by KL divergence) which satisfies a set of posterior constraints. We consider equality constraints which are in the form of $E_q[\phi(\mathbf{X}, \mathbf{Y})] = \mathbf{b}$, where $\mathbf{b}$ is set based on domain knowledge. We can also consider these constraints as features, which encode indicative patterns for a given support type label and prior beliefs on the correlations between the patterns and the true labels.

In this work, we consider two ways of generating constraints. One approach is to manually define constraints, leveraging on our domain knowledge. For instance, the unigram "should" is usually used as part of imperative, meaning it is tightly associated with the UNVERIF class. similarly, having 2 or more occurrences of a strong subjective token is also a distinguishing feature for UNVERIF. We manually define 10 constraints in this way. The other approach is to automatically extract constraints from the given labeled training data using information gain with respect to the classes as a guide.

2.3 Features

As the goal of this work is to test the efficacy of CRFs with respect to this task, most of the features are taken from the best feature combination reported in Park and Cardie (2014) for a fair comparison.

Unigrams and Bigrams. This is a set of binary features capturing whether a given unigram or bigram appears in the given proposition. N-grams are useful, because certain words are highly associated with a class. For instance, sentiment words like *happy* is associated with the UNVERIF class, as propositions bearing emotion are typically unverifiable. Also, verbs in past tense, such as *went*, is likely to appear in VERIF$_{EXP}$ propositions, because action verbs in the past tense form are often used in describing a past event in a non-subjective fashion.

Parts-of-Speech (POS) Count Based on the previous work distinguishing imaginative and informative writing, the conjecture is that the distribution of POS tags can be useful for telling apart UNVERIF from the rest (Rayson et al., 2001).

Dictionary-based Features. Three feature sets leverage on predefined lexicons to capture informative characteristics of propositions. Firstly, the subjectivity clue lexicon is used to recognize occurrences of sentiment bearing words (Wilson, 2005). Secondly, a lexicon made of speech event text anchors from the *MPQA 2.0* corpus are used to identify speech events, which are typically associated with VERIF$_{NON}$ or VERIF$_{EXP}$ (Wilson and Wiebe, 2005). Lastly, imperatives, which forms a subclass of UNVERIF, are recognized with a short lexicon of imperative expressions, such as *must, should, need to*, etc.

Emotion Expression Count The intuition is having much emotion often means the given proposition is subjective and thus unverifiable. Thus, the level of

emotion in text is approximated by counting tokens such as "!" and fully capitalized words.

Tense Count The verb tense can provide a crucial information about the type of the proposition. For instance, the future tense is highly correlated with UNVERIF, because propositions about a future event is generally unverifiable at the time the proposition is stated. Also, the past tense is a good indicator of UNVERIF or VERIF$_{EXP}$, since propositions of type VERIF$_{NON}$ are usually factual propositions irrelevant of time, such as "peanut reactions can cause death."

Person Count One example of the grammatical person being useful for classification is that VERIF$_{NON}$ propositions rarely consist of first person narratives. Also, imperatives, instances of UNVERIF, often comes with the second person pronoun.

3 Experiments and Analysis

3.1 Experiment Setup

The experiments were conducted on the dataset from Park and Cardie (2014), which consists of user comments collected from *RegulationRoom.org*, an experimental eRulemaking site. The dataset consists of user comments about rules proposed by government agencies, such as the Department of Transportation. For comparison purposes, we used the same train/test split (See Table 1). On average, roughly 8 propositions constitute a comment in both sets.

The goal of the experiments is two-fold: 1) comparing the overall performance of CRF-based approaches to the prior results from using multiclass SVMs and 2) analyzing how the semi-supervised CRFs perform with different percentages of the training data labeled, under different conditions. To achieve this, a set of repeated experiments were conducted, where gradually increasing portions of the training set were used as labeled data with the remaining portion used as unlabeled data.[4]

For evaluation, we use the macro-averaged F1 score computed over the three classes. Macro-F1 is used in the prior work, as well, to prevent the performance on the majority class[5] from dominating the overall evaluation.

	VERIF$_{NON}$	VERIF$_{EXP}$	UNVERIF	Total
Train	987	900	4459	6346
Test	370	367	1687	2424
Total	1357	1267	6146	8770

Table 1: # of Propositions in Training and Test Set

3.2 Results and Discussion

CRF vs Multiclass SVM As shown in Table 2, the multiclass SVM classifier performs better overall. But at the same time, a clear trend can be observed: With CRF, the precision makes a significant gain at the cost of the recall for both VERIF$_{NON}$ and VERIF$_{EXP}$. And the opposite is the case for VERIF.

One cause for this is the heavy skew in the dataset that can be better handled in SVMs; As mentioned before, the majority class (UNVERIF) comprises about 70% of the dataset. When training the multiclass SVM, it is relatively straight forward to balance the class distribution in the training set, as each proposition is assumed to be independent of others. Thus, Park and Cardie randomly oversample the instances of non-majority classes to construct a balanced trained set. The situation is different for CRF, since the entire sequence of propositions comprising a comment is classified together. Further investigation in resolving this issue is desirable.

Semi-supervised CRF Table 3 reports the average performance of CRFs trained on 25%, 50%, 75% and 100% labeled training data (the same dataset), using various supervised and semi-supervised approaches over 5 rounds. Though, the amount is small, incorporating semi-supervised approaches consistently boosts the performance for the most part. The limited gain in performance is due to the small set of accurate constraints.

As discussed in Section 2.2, one crucial component of training CRFs with Posterior Regularization is designing constraints on features. For a given feature, a respective constraint defines a probability distribution over the possible classes. For the best performance, the distribution needs to be accurate, and the constrained features occur in the unlabeled training set frequently.

[4] Mallet (2002) was used for training the CRFs.

[5] UNVERIF comprises about 70% of the data

Method	UNVERIF vs All			VERIF$_{NON}$ vs All			VERIF$_{EXP}$ vs All			F_1
	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1	(Macro-Ave.)
Multi-SVM (P&C)	**86.86**	83.05	84.91	49.88	**55.14**	**52.37**	66.67	**73.02**	**69.70**	**68.99**
Super-CRF 100%	80.35	**93.30**	**86.34**	**60.34**	28.38	38.60	**74.57**	59.13	65.96	63.63

Table 2: Multi-SVM vs Supervised CRF Classification Results

Method	UNVERIF vs All			VERIF$_{NON}$ vs All			VERIF$_{EXP}$ vs All			F_1
	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1	(Macro-Ave.)
Super-CRF 100%	80.35	93.30	86.34	60.34	28.38	38.60	74.57	59.13	65.96	63.63
Super-CRF 75%	79.57	92.59	85.59	54.33	30.54	39.10	77.08	53.13	62.90	62.53
CRF-PR$_H$ 75%	79.42	93.12	85.73	57.14	31.35	40.49	79.01	52.32	62.95	63.06
CRF-PR$_{H+IG}$ 75%	79.72	94.37	86.43	63.58	27.84	38.72	76.6	55.31	64.24	63.13
Super-CRF 50%	79.16	93.01	85.53	51.92	21.89	30.82	71.68	55.86	62.79	59.71
CRF-PR$_H$ 50%	79.28	92.12	85.17	55.68	26.49	35.92	69.23	53.95	60.64	60.57
CRF-PR$_{H+IG}$ 50%	79.23	92.23	85.24	55.37	26.49	35.83	70.32	54.22	61.23	60.77
Super-CRF 25%	75.93	96.86	85.13	57.89	5.95	10.78	79.06	50.41	61.56	52.49
CRF-PR$_H$ 25%	76.27	96.03	85.02	41.54	7.30	12.41	79.15	50.68	61.79	53.07
CRF-PR$_{H+IG}$ 25%	75.83	96.32	84.86	38.78	5.14	9.07	79.31	50.14	61.44	51.79

Table 3: Supervised vs Semi-Supervised CRF Classification Results

*The percentages refer to the percentages of the labeled data in the training set.
*The methods are as follows: Super-CRF = supervised approach only using the labeled data, CRF-PR$_H$ = CRF with posterior regularization using constraints that are manually selected, CRF-PR$_{H+IG}$ = CRF with posterior regularization using constraints that are manually written and automatically generated using information gain.
*Precision, recall, and F_1 scores are computed with respect to each one-vs-all classification problem for evaluation purposes, though a single model is built for the multi-class classification problem.

Our manual approach resulted in a small set of about 10 constraints on features that are tightly coupled with a class. Examples include the word "should", large number of strong subjective expressions, and imperatives, which are all highly correlated with the UNVERIF. While the constraints are accurate, the coverage is too small to boost the performance. However, it is quite difficult to generate a large set of constraints, because there are not that many features that are indicative of a single class. Also, given that UNVERIF comprises a large percentage of the dataset, and the nature of verifiability[6], it is even more difficult to identify features tightly coupled with VERIF$_{NON}$ and VERIF$_{EXP}$ class. One issue with automatically generated constraints, based on information gain, is that they tend to be inaccurate.

[6]Verifiability does not have many characterizing features, but the lack of any of the characteristics of unverifiability, such as sentiment bearing words, is indicative of verifiability.

4 Conclusions and Future Work

We present an empirical study on employing Conditional Random Fields for identifying appropriate types of support for propositions in user comments. An intuitive extension to Park and Cardie (2014)'s approach is to frame the task as a sequence labeling problem to leverage on the fact that certain types of propositions tend to occur together. While the overall performance is reduced, we find that Conditional Random Fields (CRFs) improves the F_1 score with respect to the UNVERIF class. Also, semi-supervised CRFs with posterior regularization trained on 75% labeled training data can closely match the performance of a supervised CRF trained on the same training data with the remaining 25% labeled as well.

An efficient way to handle the skewed distribution of classes in the training set is needed to boost the performance of CRFs. And a set of efficient constraints is necessary for better performing semi-supervised CRFs with posterior regularization.

References

Kevin D. Ashley and Vern R. Walker. 2013. From information retrieval (ir) to argument retrieval (ar) for legal cases: Report on a baseline study. In *JURIX*, pages 29–38.

Kedar Bellare, Gregory Druck, and Andrew McCallum. 2009. Alternating projections for learning with expectation constraints. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 43–50. AUAI Press.

Elena Cabrio and Serena Villata. 2012. Combining textual entailment and argumentation theory for supporting online debates interactions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 208–212, Jeju Island, Korea, July. Association for Computational Linguistics.

Vanessa Wei Feng and Graeme Hirst. 2011. Classifying arguments by scheme. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 987–996, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kuzman Ganchev, Jennifer Gillenwater, and Ben Taskar. 2009. Dependency grammar induction via bitext projection constraints. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 369–377. Association for Computational Linguistics.

Kuzman Ganchev, Joao Graça, Jennifer Gillenwater, and Ben Taskar. 2010. Posterior regularization for structured latent variable models. *The Journal of Machine Learning Research*, 99:2001–2049.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://www.cs.umass.edu/ mccallum/mallet.

Raquel Mochales Palau and Marie-Francine Moens. 2009. Argumentation mining: The detection, classification and structure of arguments in text. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law*, ICAIL '09, pages 98–107, New York, NY, USA. ACM.

Joonsuk Park and Claire Cardie. 2014. Identifying appropriate support for propositions in online user comments. In *Proceedings of the First Workshop on Argumentation Mining*, pages 29–38, Baltimore, Maryland, June. Association for Computational Linguistics.

Paul Rayson, Andrew Wilson, and Geoffrey Leech. 2001. Grammatical word class variation within the british national corpus sampler. Language and Computers.

Maria Paz Garcia Villalba and Patrick Saint-Dizier. 2012. Some facets of argument mining for opinion analysis. In *COMMA*, pages 23–34.

Theresa Wilson and Janyce Wiebe. 2005. Annotating attributions and private states. In *Proceedings of ACL Workshop on Frontiers in Corpus Annotation II: Pie in the Sky*.

Theresa Wilson. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *In Proceedings of HLT-EMNLP*, pages 347–354.

Adam Wyner, Raquel Mochales-Palau, Marie-Francine Moens, and David Milward. 2010. Semantic processing of legal texts. chapter Approaches to Text Mining Arguments from Legal Cases, pages 60–79. Springer-Verlag, Berlin, Heidelberg.

Bishan Yang and Claire Cardie. 2014. Context-aware learning for sentence-level sentiment analysis with posterior regularization. In *Proceedings of ACL*.