Is Something Better than Nothing? Automatically Predicting Stance-based Arguments using Deep Learning and Small Labelled Dataset.

Pavithra Rajendran[1], Danushka Bollegala[1], Simon Parsons[2]
University of Liverpool[1], Kings College London[2]

Abstract

Online reviews have become a popular portal among customers making decisions about purchasing products. A number of corpora of reviews have been widely investigated in NLP in general, and, in particular, in argument mining. This is a subset of NLP that deals with extracting arguments and the relations among them from user-based content. A major problem faced by argument mining research is the lack of human-annotated data. In this paper, we investigate the use of weakly supervised and semi-supervised methods for automatically annotating data, and thus providing large annotated datasets. We do this by building on previous work that explores the classification of opinions present in reviews based on whether the stance is expressed explicitly or implicitly. In the work described here, we automatically annotate stance as implicit or explicit and our results show that the datasets we generate, although noisy, can be used to learn better models for implicit/explicit opinion classification.

1 Introduction

Sentiment analysis and opinion mining are widely researched NLP sub-fields that have extensively investigated opinion-based data such as online reviews (Pang et al., 2008; Cui et al., 2006). Reviews contain a wide range of opinions posted by users, and are useful for customers in deciding whether to buy a product or not. With abundant data available online, analysing online reviews becomes difficult, and tasks such as sentiment analysis are inadequate to identify the reasoning behind a user's review. Argument mining is an emerging research field that attempts to solve this problem by identifying arguments and the relation between them using ideas from argumentation theory (Palau and Moens, 2009).

An argument can be defined in two different ways – (1) abstract arguments which need not have any internal structure (Dung, 1995) and (2) structured arguments where an argument is a collection of premises leading to a conclusion. One major problem that is faced by argument mining researchers is the variation in the definition of an argument, which is highly dependent on the data at hand. Previous works in argument mining has mostly focussed on a particular domain (Grosse et al., 2015; Villalba and Saint-Dizier, 2012; Ghosh et al., 2014; Boltuzic and Snajder, 2014; Park and Cardie, 2014; Cabrio and Villata, 2012). Furthermore, an argument can be defined in a variety of ways depending on the problem being solved. As a result, we focus on the specific domain of opinionated texts such as those found in online reviews.

Prior work (Carstens et al., 2014; Rajendran et al., 2016a) in identifying arguments in online reviews have considered sentence-level statement as arguments based on abstract argumentation models that is relatively easier to achieve. However, to extract arguments at a finer level based on the structured argument definition requires us to manually annotate argument components such that they can be used in supervised techniques. Because of the heterogenous nature of user-based contents, this task is time-consuming and expensive (Khatib et al., 2016; Habernal and Gurevych, 2015) and often domain-dependent.

In this work, we are interested in analysing the problem where human-annotated or labelled data is small in size and how it can be overcome using weakly-supervised and semi-supervised techniques. We consider one such particular work (Rajendran et al., 2016b), which manually annotates a small dataset for a supervised binary classification on opinions present in online reviews, based on how the stance is expressed linguistically in the

Opinion	Stance	Aspect	Annotation
Great *hotel*!	direct	hotel	Explicit
don't get fooled by book reviews and movies, this *hotel* is not a five star luxury experience, it dosen't even have sanitary standards!	direct and indirect	hotel	Explicit
another annoyance was the *internet* access, for which you can buy a card for 5 dollars and this is supposed to give you 25 mins of access, but if you use the card more than once, it debits an access charge and rounds minutes to the nearest five.	indirect	internet	Implicit
the other times that we contacted front desk/guest services (very difficult to tell them apart) we were met by unhelpful unknowledgable staff for very straightforward requests verging on the sarcastic and rude	indirect	staff	Implicit
the attitude of all the staff we met was awful, they made us feel totally unwelcome	direct and indirect	staff	Explicit

Table 1: Examples of opinions along with the following information: whether the stance is directly (and) or indirectly expressed, the aspect present and whether the opinion is annotated explicit or implicit.

structure of these opinions. One disadvantage of their work is the lack of large labelled data but we do have a large amount of unannotated (unlabelled) online reviews written by reviewers at our disposal (e.g. TripAdvisor [1]).

Our motivation is to investigate on whether automatically labelling a large set of unlabelled opinions as implicit/explicit can assist learning deep learning models for the implicit/explicit classification task and also for other related tasks that depend on this classification. In our investigation, we are interested in automatically labelling such a dataset using the previously proposed supervised approach described in (Rajendran et al., 2016b).

Experiments are carried out using two different approaches – weakly-supervised and semi-supervised learning (Section. 4). In the weakly-supervised approach, we randomly divide the manually annotated implicit/explicit opinions into different training sets that are used to train SVM classifiers for automatically labelling unannotated opinions. The unannotated opinions are labelled based on different voting criteria — *Fully-Strict*, *Partially-Strict* and *No-Strict*. In the semi-supervised approach, an SVM classifier is either trained on a portion of the annotated implicit/explicit opinions or using the entire data. The resulting classifier is then used to predict the unannotated opinions and those with highest confidence are appended to the training data. This process is repeated for m iterations.

All the approaches give us a set of automatically labelled opinions. A Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) model is trained on this data and tested on the original manually-annotated dataset. Results show that the maximum overall accuracy of 0.84 on the annotated dataset is obtained using an LSTM model trained using the labelled data generated by the weakly-supervised approach using the *Partially-Strict* voting criterion.

[1] www.tripadvisor.com

2 Related work

Research in argument mining attempts to automatically identify arguments and their relations that are present in natural language texts. Lippi and Torroni (2016) present a detailed survey on the existing work in argument mining. These are carried out on different domains such as debates (Cabrio and Villata, 2012; Habernal and Gurevych, 2016), reviews (Wyner et al., 2012; Gabbriellini and Santini, 2015), tweets (Bosc et al., 2016), dialogues (Biran and Rambow, 2011) etc. Amgoud et al. (2015) find arguments in such texts as not formally structured with most of the content left implicit. An argument, in general, is treated as a set of premises or reasoning that are linked to a claim or conclusion and, those arguments in which the major premises are left missing are termed as enthymemes. It is important to understand whether the implicitly left content in natural language texts are to be dealt as enthymemes or not. In our earlier work (Rajendran et al., 2016b), we propose an approach for reconstructing structures similar to enthymemes in opinions that are present in online reviews. However, the annotated dataset used in our approach was small and not useful for modelling deep learning models. Recent work in argument mining are able to achieve a better performance for the argument identification task using neural network models with the availability of a large corpus of annotated arguments (Habernal et al., 2018; Eger et al., 2017). Annotating a large corpus is a tedious task and few existing work in argument mining have explored alternative ways

to do it. Naderi and Hirst (2014) proposes a frame-based approach for dealing with arguments present in parliamentary discourse and suggests that using a semi-supervised approach can help in developing their dataset into a large corpus. Habernal and Gurevych (2015) have proposed a semi-supervised based approach for identifying arguments using a clustering based approach on unlabelled data. Their results outperform several baselines and provide a way of developing their corpus without having to manually annotate the entire dataset. In this paper, we show that a small labelled dataset trained using an existing SVM-based classifer with the best features can help in automatically labelling a large dataset and we also evaluate its usefulness for modelling deep learning models.

3 Implicit/Explicit classification

Prior work (Rajendran et al., 2016b) defines a sentence-level statement that is of a positive/negative sentiment and talks about a target as a stance-containing opinion. Biber and Finegan (1988) defines stance as the expression of the user's attitude and judgement in their message to convince the audience towards the standpoint taken by them. This is different from the definition used for stance detection in NLP, in which, a given piece of text is classified as being for or against a given claim. Based on the definition given in Biber and Finegan (1988), stance-containing opinions are classified as being implicit/explicit based on how the stance or the standpoint of the reviewer towards the target is being expressed in the linguistic structure of the opinion. This definition of what we term as explicit or implicit may depend on the audience interpretation and may vary for evey individual. In order to make the human annotation task less tedious, Rajendran et al. (2016b) use the following cues to label the opinions as implicit or explicit. These opinions are extracted from hotel reviews present in the ArguAna corpus (Wachsmuth et al., 2014). Some examples from Rajendran et al. (2016b) are given in Table. 1.

Explicit opinion

1. Direct approval/disapproval is expressed by the reviewer. Few examples are: *I do not like the hotel*, *I would definitely recommend this hotel*
2. Strong intensity of expression. Certain words or clauses have a strong positive/negative intensity towards a particular target. For example, *worst staff!* has a strong negative intensity in comparison to *the staff were not helpful*.

Implicit opinion

1. Words or clauses indicate positive/negative expression but do not express it with a strong intensity. For example, *the staff were friendly and helped us with our baggages.*
2. Opinions that are expressed as personal facts. Few examples are *small room*, *carpets are dirty* etc.
3. Opinions that express a form of justification such as describing an incident that indirectly is meant to imply the reviewer's satisfaction or dissatisfaction. For example, *they made us wait for a long time for the check-in and the staff completely ignored us.*

To overcome the data imbalance for the two classes, the original dataset annotated by a single annotator was undersampled in (Rajendran et al., 2016b) into 1244 opinions (495 explicit and 749 implicit). Next, two annotators were asked to independently annotate this undersampled dataset, and the inter-annotator agreement for this task is 0.70, measured using Cohen's κ (Cohen, 1960).

4 Methodology

4.1 Weakly-supervised Approach

Our first experiment uses a method that is similar to bagging (Breiman, 1996). Starting from a randomly selected subset of the undersampled annotated data, we first create three different training sets, T_1, T_2 and T_3. These training sets are then each used to train an SVM classifier which uses the highest discriminative features (Rajendran et al., 2017) identified for predicting implicit and explicit stance:

Unigrams and Bigrams Each word present in an opinion and each consecutive words present in an opinion is considered as features.

Noun-Adjective pattern Let us consider $\mathcal{N}$ to represent the list of nouns and $\mathcal{A}$ to represent the list of adjectives in an opinion. The combination of each noun with an adjective is considered as a Noun tag + Adjective tag

Dataset	Labelled Data		Average-based		Fully-Strict		Partially-Strict		No-Strict	
	Exp	Imp	Size	Acc	Size	Acc	Size	Acc	Size	Acc
D1	100	749	4931	73.95	4376	72.99	4541	75.56	4931	67.76
D2	200	749	4931	79.5	4310	75.64	4575	82.07	4931	71.66
D3	300	749	4931	80.99	4427	79.50	4655	83.36	4931	73.71
D4	400	749	4931	81.50	4541	78.13	4726	84.08	4931	76.36
D5	495	100	4931	76.41	3411	76.20	4113	75.32	4931	82.23
D6	495	200	4931	81.72	3742	83.52	4276	80.30	4931	83.19
D7	495	300	4931	83.01	4054	83.36	4409	83.44	4931	79.90
D8	495	400	4931	82.42	4054	83.60	4498	84.08	4931	82.31
D9	495	500	4931	83.54	4501	83.44	4762	84.00	4931	82.63
D10	495	600	4931	83.75	4484	83.52	4762	83.52	4931	82.39
D11	495	700	4931	82.15	4678	83.19	4797	84.00	4931	82.55

Table 2: Datasets vary in the number of explicit and implicit opinions that are randomly sampled from the labelled data to be trained by the SVM classifier. For each of the weakly supervised approach, we give *size*, the number of the predicted labels that are used to train an LSTM-based model. This model was then tested on the entire labelled data, and the accuracy of this LSTM model is reported.

feature.

$$\mathrm{C} = \sum_{i=1}^{k} \sum_{j=1}^{l} NN + JJ \qquad (1)$$

where k is the total number of nouns present and l is the total number of adjectives present.

Average-based sentence embedding We compute the mean of the 300-dimensional pre-trained word embedding vectors trained using GloVe (Pennington et al., 2014) to create a sentence embedding, and use each dimension in the sentence embedding as a feature in the classifier.

$$\mathbf{v} = \frac{1}{|\mathrm{S}|} \sum_{i=1}^{|\mathrm{S}|} \mathbf{s}_i \qquad (2)$$

where $|\mathbf{S}|$ represents the size of the opinion and $\mathbf{s}_i$ represents the pre-trained word embedding for the $i-th$ word in the opinion.

The three resulting SVM classifiers are then used to annotate 4931 unannotated opinions, and these newly annotated opinions are then used to train an LTSM classifier. We generate the annotated opinions in two different ways — what we call the average-based method and the voting-based method — and for each method we use the resulting annotated opinions differently as described next.

Average-Based Each training set T_1, T_2 and T_3 is used to train separate SVM classifiers, which are used to label the unlabelled opinions, giving corresponding annotated opinion sets U_1, U_2 and U_3. Separate LSTM models are trained on each of U_1, U_2 and U_3, and tested on the original set of annotated data. Finally, the averaged performance across the three LSTMs is reported.

Voting-Based Again, each training set T_1, T_2 and T_3 is used to train separate SVM classifiers, which are used to label the unlabelled opinions, giving corresponding annotated opinion sets U_1, U_2 and U_3. We then followed an approach that is similar to Ng and Cardie (2003) to combine the opinions in U_1,U_2 and U_3 into a single set, denoted by U_F, using the following voting criteria:

Fully-Strict An opinion is included in U_F if all three SVM classifiers predict the same stance label.

Partially-Strict An opinion is included in U_F if all three SVM classifiers identify it as explicit, or if at least two of them classify it as implicit.

No-Strict An opinion is included in U_F as implicit if at least one of the classifiers predict it to be implicit, otherwise it is included in U_F as explicit.

U_F was then used to train an LSTM classifier and this was tested on the original annotated data.

Note that moving from Fully-strict → Partially-Strict → No-Strict relaxes the requirement on including an opinion in U_F so that the number of opinions in the training data increases.

4.2 Semi-supervised approach

We conduct a second experiment to test the combination of both labelled (1244 opinions) and unlabelled (4931 opinions) data using the following popular semi-supervised learning methods.

Iterations	Self-training		Reserved	
	Size	Accuracy	Size	Accuracy
1	22	49.43	511	67.68
5	2110	80.86	1717	68.24
10	2574	81.83	2194	70.25
15	3600	82.71	3152	70.98
20	3613	82.71	3708	68.81
25	4931	82.71	4931	64.22

Table 3: Accuracy of the LSTM model on annotated data using a set of automatically labelled unannotated opinions of *Size*.

Self-training method We train an SVM using the labelled data D and use this to annotate the unannotated data U. The annotated opinions from U which are labelled with the highest probability are then added to D. This process is repeated m times.

Reserved method Here we use the method of Liu et al. (2013), where a portion of the training data R is reserved, and the remainder is used for training the SVM. The resulting classifier is run on the combination of U and R. The annotated opinions from U with the highest probability and the opinions from R that have the lowest probability of having a correct label generated by the SVM are appended to the training dataset. This operation is repeated m times. We chose 222 explicit opinions and 287 implicit opinions as the training data, and took 273 explicit opinions and 462 implicit opinions as the reserved portion.

After the final iteration, the final set of annotations of the opinions in U is used to train an LSTM model. The resulting classifier is then tested on the original set of annotated data.

5 Experiment and Results

We used Keras[2] to implement an LSTM model with an embedding layer using pre-trained 300 dimensional GloVe embeddings, followed by an LSTM layer of size 100 with a dropout rate of 0.5 and a sigmoid output layer. The input length is padded to 50. Parameter optimisation is done using Adam (Kingma and Ba, 2014). For the semi-supervised approaches, we consider the number of iterations, $m = 1 - 25$.

Table. 2 reports under *Size* the number of unannotated data that is automatically labelled using the weakly-supervised approaches. The corresponding columns *Exp* and *Imp* contain the number of manually annotated opinions that are used to train the SVM classifier used in the first-step of the proposed method. The *Acc* column denotes the accuracy for predicting the labels of the annotated dataset using the LSTM model trained on the automatically labelled, unannotated data.

Looking at the performance of the weakly-supervised approach in Table. 2, we observe the varying the size of the explicit and the implicit opinions that are used to train the SVM-based classifier (see columns *Emp* and *Imp* in Table. 2) and compare them with the accuracy scores, we find that using the largest set of explicit opinions in training the initial SVMs gives new annotated data that can train classifiers that perform best on the original annotated data. Overall, using the entire undersampled data for training the SVMs and using the *Partially-Strict* voting based method gives the best performance with an accuracy of 0.84.

Table. 3 reports the results obtained using the self-training method and the reserved method. These show how the size of the labelled unannotated dataset increases at each iteration and these newly annotated opinions are added to the training data. The accuracy of the LSTM model in predicting the labels of annotated opinions improves with the size of the automatically labelled dataset. However, the accuracy of the reserved method decreases in performance after 20 iterations[3]. Of the two methods, the self-training method performs best, showing that using training data with lowest confidence does not help in this task.

Overall, the results are positive, showing a range of methods that can create automatically labelled data which is accurate enough to be useful for deep-learning methods.

The dataset is made publicly available at https://goo.gl/Bym2Vz.

6 Conclusion

This work investigated a particular task related to argument mining where we have a small annotated dataset. Our results show that using a semi-supervised method with the available small annotated dataset is sufficient to label a larger unlabelled dataset so it can be used to train a deep learning LSTM model for the argument mining task.

[2] https://keras.io/

[3] This is typical of such methods as less reliable examples are added to the training data.

References

Leila Amgoud, Philippe Besnard, and Anthony Hunter. 2015. Representing and reasoning about arguments mined from texts and dialogues. In *ECSQARU*. pages 60–71.

Douglas Biber and Edward Finegan. 1988. Adverbial stance types in english. In *Discourse Processes*. volume 11, pages 1–34.

Or Biran and Owen Rambow. 2011. Identifying justifications in written dialogs. In *ICSE*. pages 162–168.

Filip Boltuzic and Jan Snajder. 2014. Back up your stance: Recognizing arguments in online discussions. In *ACL*. pages 49–58.

Tom Bosc, Elena Cabrio, and Serena Villata. 2016. Dart: a dataset of arguments and their relations on twitter. In *LREC*. pages 1258–1263.

Leo Breiman. 1996. Bagging predictors. *Machine learning* 24(2):123–140.

Elena Cabrio and Serena Villata. 2012. Combining textual entailment and argumentation theory for supporting online debates interactions. In *ACL*. pages 208–212.

Lucas Carstens, Francesca Toni, and Valentinos Evripidou. 2014. Argument mining and social debates. In *COMMA*. pages 451–452.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement* 20(1):37–46.

Hang Cui, Vibhu Mittal, and Mayur Datar. 2006. Comparative experiments on sentiment classification for online product reviews. In *AAAI*. pages 1265–1270.

P. M. Dung. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.* 77:321–357.

Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. 2017. Neural end-to-end learning for computational argumentation mining. In *ACL*. pages 11–22.

Simone Gabbriellini and Francesco Santini. 2015. A micro study on the evolution of arguments in amazon. coms reviews. In *PRIMA*. pages 284–300.

Debanjan Ghosh, Smaranda Muresan, Nina Wacholder, Mark Aakhus, and Matthew Mitsui. 2014. Analyzing argumentative discourse units in online interactions. In *ACL*. pages 39–48.

Kathrin Grosse, Maria P Gonzalez, Carlos I Chesnevar, and Ana G Maguitman. 2015. Integrating argumentation and sentiment analysis for mining opinions from twitter. *AI Communications* 28:387–401.

Ivan Habernal and Iryna Gurevych. 2015. Exploiting debate portals for semi-supervised argumentation mining in user-generated web discourse. In *EMNLP*. pages 2127–2137.

Ivan Habernal and Iryna Gurevych. 2016. Which argument is more convincing? analyzing and predicting convincingness of web arguments using bidirectional lstm. In *ACL*. pages 1589–1599.

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2018. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. In *NAACL-HLT*. page (to appear).

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Khalid Al Khatib, Henning Wachsmuth, Matthias Hagen, Jonas Kohler, and Benno Stein. 2016. Cross-domain mining of argumentative text through distant supervision. In *NAACL-HLT*. pages 1395–1404.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR* abs/1412.6980.

Marco Lippi and Paolo Torroni. 2016. Argumentation mining: State of the art and emerging trends. *TOIT* 16(2):10.

Zhiguang Liu, Xishuang Dong, Yi Guan, and Jinfeng Yang. 2013. Reserved self-training: A semi-supervised sentiment classification method for chinese microblogs. In *IJCNLP*. pages 455–462.

Nona Naderi and Graeme Hirst. 2014. Argumentation mining in parliamentary discourse. In *PRIMA*. pages 16–25.

Vincent Ng and Claire Cardie. 2003. Weakly supervised natural language learning without redundant views. In *NAACL-HLT*. pages 94–101.

Raquel Mochales Palau and Marie-Francine Moens. 2009. Argumentation mining: the detection, classification and structure of arguments in text. In *ICAIL*. pages 98–107.

Bo Pang, Lillian Lee, et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval* 2(1–2):1–135.

Joonsuk Park and Claire Cardie. 2014. Identifying appropriate support for propositions in online user comments. In *ACL*. pages 29–38.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*. pages 1532–1543.

Pavithra Rajendran, Danushka Bollegala, and Simon Parsons. 2016a. Assessing weight of opinion by aggregating coalitions of arguments. In *COMMA*. pages 431–438.

Pavithra Rajendran, Danushka Bollegala, and Simon Parsons. 2016b. Contextual stance classification of opinions: A step towards enthymeme reconstruction in online reviews. In *ArgMining@ACL*. pages 32–39.

Pavithra Rajendran, Danushka Bollegala, and Simon Parsons. 2017. Identifying argument based relation properties in opinions. In *PACLING*.

Maria Paz Garcia Villalba and Patrick Saint-Dizier. 2012. A framework to extract arguments in opinion texts. *IJCINI* 6:62–87.

Henning Wachsmuth, Martin Trenkmann, Benno Stein, Gregor Engels, and Tsvetomira Palakarska. 2014. A review corpus for argumentation analysis. In *CICLing*. pages 115–127.

Adam Wyner, Jodi Schneider, Katie Atkinson, and Trevor J. M. Bench-Capon. 2012. Semi-automated argumentative analysis of online product reviews. In *COMMA*. pages 43–50.