2015 IEEE 15th International Conference on Data Mining Workshops

Improving Out-of-domain Sentiment Polarity Classification using Argumentation

Lucas Carstens
Imperial College London
180 Queens Gate
SW7 2AZ
London, United Kingdom
Email: lc1310@imperial.ac.uk

Francesca Toni
Imperial College London
180 Queens Gate
SW7 2AZ
London, United Kingdom
Email: ft@imperial.ac.uk

***Abstract*—Domain dependence is an issue that most researchers in corpus-based computational linguistics have faced at one time or another. With this paper we describe a method to perform sentiment polarity classification across domains that utilises Argumentation. We train standard supervised classifiers on a corpus and then attempt to classify instances from a separate corpus, whose contents are concerned with different domains (e.g. sentences from film reviews vs. Tweets). As expected the classifiers perform poorly and we improve upon the use of a simple classifier for out-of-domain classification by taking class labels suggested by classifiers and *arguing* about their validity. Whenever we can find enough arguments suggesting a mistake has been made by the classifier we change the class label according to what the arguments tell us. By arguing about class labels we are able to improve $F1$ measures by as much as 14 points, with an average improvement of $F1 = 7.33$ across all experiments.**

I. Introduction

Few phenomena are as pervasive in language as sentiment; we can find opinionated text in virtually any domain we may be interested in. Accordingly, Sentiment Analysis has been met with broad interest in the Natural Language Processing (NLP) community and many annotated corpora have been developed (see e.g. [1], [2], [3]). Nevertheless domain dependence remains a crucial challenge for Sentiment Analysis, as it does for many other NLP problems. A classification model built using a corpus of annotated product reviews may work well on text taken from Amazon, but its performance may drop off steeply when we try to use it to classify Tweets. One way of addressing issues of domain dependence is by building classification models that are tailored to specific domains. The more narrow the scope of a model, and the corpus it is trained on, is, the better we may expect the model to perform when classifying text from this domain. In tailoring a model to a certain domain, however, we not only get the benefit of increased performance within this domain, we also risk sacrificing performance when trying to classify text that is not strictly part of the domain in question. Additionally we increase the amount of work needed to build our domain-specific models, since, for each model, we need to have available a domain-specific corpus.

With this paper we describe a method of increasing the performance of out-of-domain text classification, in the hope of decreasing the need for constructing new corpora for each domain we are interested in. Specifically, we address the problem of sentiment polarity classification. We combine generic classification models, such as *Random Forests (RF)* [4], trained on a domain-specific corpus, with a set of domain-independent *Arguments*, to classify out-of-domain instances. Each Argument asserts that an instance is either positive or negative. If we can find sufficient arguments that disagree with the class label provided by the classifier, we overturn the classification decision of the classifier and change it to what the Arguments tell us the classification should be.

The remainder of this paper is organised as follows: We first review relevant related work from the fields of Sentiment Analysis and Argumentation in section II. In section III we describe our approach to integrating domain-independent knowledge using Argumentation to perform sentiment polarity classification. Based on this, in section IV we discuss the two-step approach through which we reach the classification of instances as either positive or negative. While step (1), described in section IV-A yields an initial class label returned by a classifier, in step (2), described in section IV-B, we take the suggested class label and argue about its validity, possibly changing the class label. In section V we review the contents of the three corpora used to for testing. We use these corpora throughout our experiments, which we describe in section VI. We discuss the results of our experiments, as well as some other relevant issues, in section VII. We conclude our paper in section VIII with a reflection on the work presented in this paper and a few tasks to be considered in our subsequent research.

II. Related work

As we explain in detail in section III, we use Argumentation to integrate domain-independent knowledge with a sentiment classifier to improve out-of-domain polarity classification. Below we give an overview of related work in Sentiment Analysis and Argumentation.

A. Sentiment Analysis

With our work we touch upon two issues that have received considerable attention from the NLP community, sentiment polarity classification and the classification of out-of-domain instances, in general. For more general surveys on Sentiment Analysis see e.g. [5], [6].

978-1-4673-8493-3/15 $31.00 © 2015 IEEE
DOI 10.1109/ICDMW.2015.185

1) Polarity classification: Most broadly Sentiment Analysis needs to address two challenges. On the one hand we need to determine whether a piece of text is *sentimental* and, on the other hand, we need to determine the polarity of text that is indeed deemed sentimental. In the work we present here we focus on determining the polarity of text and will hence review related work addressing this particular challenge.

Polarity classification has been addressed on various levels of granularity, e.g. on word, sentence and text level. Wilson and colleagues [3] analyse phrases, taken from the *Multi Purpose Question Answering (MPQA)* corpus [7], both according to whether they are polar or neutral and whether the polar sentences are positive or negative. To classify phrases they develop a subjectivity lexicon of around $8,000$ *subjectivity clues*, which is built upon an earlier lexicon proposed in [8]. In addition to identifying subjectivity clues they define features pertinent to identifying sentiment that are based on the structure of the sentence a phrase appears in, the topic of the document a phrase appears in, etc. We discuss more features that have been used in Sentiment polarity classification in section VIII-B

Pang and Lee have conducted extensive work on identifying sentiment in film reviews on various levels of granularity [9], [10], [11]. They present work both on identifying the polarity of sentences and entire film reviews. We describe the corpus developed for sentence polarity classification in section V-C, as we use this particular corpus for our experiments, summarised in section VI. Twitter has also been a popular subject, with various researchers tackling polarity classification on Tweets [12], [13], [14], [15].

2) Out-of-domain classification: The identification of sentiment across domains has been a prominent topic in the field. Xia and colleagues [16] propose the *feature ensemble plus sample selection (SS-FE)* method; they also provide a good overview of domain adaptation work in Sentiment Analysis. They create ensembles of features based on how domain-dependent they appear to be, determining features whose sentiment carries across domains and those whose sentiment changes across domains. Sample selection refers to determining a useful subset of available annotated instances based on which the feature ensembles are built.

Much work has focused on developing resources such as thesauri that hold sentiment laden words whose polarity or strength are as generally valid as possible. Li and colleagues [17] use what they call knowledge transformation to broaden the applicability of data that is already annotated. To do so they extract reviews from sources that provide meta data, such as the *Internet Movie Database (IMDB)* and try to extract terms that convey sentiment beyond the setting in which they are used in that particular dataset. Bollegala and colleagues [18] similarly propose the development of a thesaurus of sentimental words that are not specific to a certain domain.

Weichselbraun and colleagues [19], [20] describe *Contextualised sentiment analysis*, identifying and resolving ambiguous terms according to the domain in which they are used. They take domain-specific corpora and extract features from them that translate well across domains. In addition, common sense knowledge bases are consulted to identify term meaning on concept-level. Tsai and colleagues [21] build a concept-level sentiment lexicon based entirely on common sense knowledge. Pan and colleagues propose *Spectral Feature Alignment (SFA)* [22]. In SFA domain words that are sentiment laden in one domain are linked to words that are sentiment laden in other domains by connecting them via words that are sentiment laden irrespective of domain.

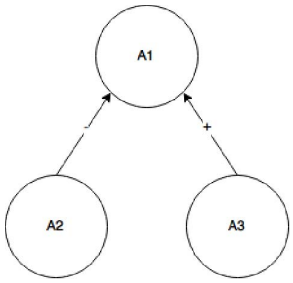

Fig. 1. Relations between arguments in a simple Argument graph, where A2 attacks A1 and A3 supports A1. Example bipolar argumentation framework (- stands for attack, + stands for support.)

B. Argumentation

Argumentation frameworks (AFs), a non-monotonic reasoning paradigm that consists of a set of arguments and relations between these arguments, have attracted considerable research attention in recent years (see [23] for an overview). AFs can be naturally represented as directed graphs, with each node representing an argument and each arc representing an attack [24]. A *Bipolar Abstract Argumentation Framework* (BAF) [25] is an AF extended with a binary *support* relation between arguments. Formally, a BAF is a triple $(Args, Attack, Support)$ where Args is a set and $Attacks/Supports \subseteq Args \times Args$ are binary relations ($(A, B) \in Support$ is read 'A *supports* B'). A BAF can then be represented as a directed graph, in which each node corresponds to an argument and each directed arc corresponds to an attack or a support (arcs need to be labelled accordingly). Take, for example, the below excerpt of a discussion between John, Joe and Jane on whether or not they should go and watch the latest *Avengers* movie in the cinema:

> John: I think we should go and see the new Avengers; the first one was great! (A1)
>
> Joe: Please spare me! It's just going to be another big Hollywood production that goes for explosions instead of plot and characters. (A2)
>
> Jane: I loved the first one, as well, so I think we should see it! (A3)

By identifying that Joe disagrees with (attacks) John and Jane agrees with (supports) John, this dialogue can be mapped into the BAF shown in Figure 1. There are multiple criteria for selecting 'winning' arguments in a BAF, which are known as *semantics* [24], [25]. Some of these semantics are defined as 'rationally acceptable' extensions. Here, however, we focus on a class of *quantitative semantics*, assessing the 'dialectical' strength of arguments numerically. In particular, we focus on two of these semantics, given in [26] and in [27] respectively, both building upon [28]. These approches are referred to as QuAD (for Quantitative Argumentation Debate) [26] and

ESAA (for Extended Social Abstract Argumentation) [27]. Both QuAD and ESAA assume that arguments are equipped with a *base score*, namely a number in the $[0, 1]$ interval. ESAA also assumes that positive and negative votes may be ascribed to arguments, that result in a modification of their base score (see [27] for details). In both approaches, the (given or modified) base score amounts to an intrinsic (non-dialectical) strength of arguments. Both approaches determine the (dialectical) strength of arguments by aggregating the strength of attackers against and supporters for these arguments, for restricted types of BAFs in the form of trees.

III. Sentiment Arguments

A major issue in developing solutions that classify texts across domains is the fact that words, phrases etc. can mean very different things, depending on the setting in which they are invoked. When we talk about electronic gadgets the word *compact* may often carry positive meaning. Encounter this word in an advertisement for a flat rental, however, and you will be keenly aware that the advertised dwelling is likely little more than a storage closet. Other words, phrases, etc., however, carry their sentiment across domains more successfully. Words such as *awful* or *unbearable* will seldom be used to invoke a positive sentiment. In order to integrate such, more domain independent, knowledge with a standard supervised classifier, we formalise it as arguments. An argument is comprised of one or more premises and a conclusion. For our purposes, premises are characteristics of sentences that indicate a certain polarity, while the conclusion is that, since a sentence contains such an indicator, it is either of positive or negative polarity. Additionally, each argument has assigned to it a certain score, equivalent to the base score in the ESAA framework. The score of an argument reflects the impact an argument should have on determining whether a sentence's polarity is positive or negative. How we may determine an argument's base score is an on-going topic of our research. Currently we form groups of arguments that share certain characteristics, e.g. have the same conclusion, and assign the same score to each argument in a group. We determine these group scores by maximising the classification accuracy on test sets maintained separately from the corpora described in section V. We discuss other ways of identifying the scores for arguments in section VIII. An argument takes the following form:

Premise(s) Conclusion Score

To exemplify, consider the following *negative* Tweet T_1, taken from the STS Twitter corpus [29], described in section V-B:

> "more depressed than you could ever imagine that I wont be going to Vegas. I hate having to be financially responsible."

An argument Arg_1 applicable to this Tweet may then go as follows:

hate negative 0.4

where the premise is that the sentence in question contains the keyword *hate* and that this is an indicator to conclude that the sentence has a negative polarity. In the following section we describe how we integrate such arguments in a classification procedure.

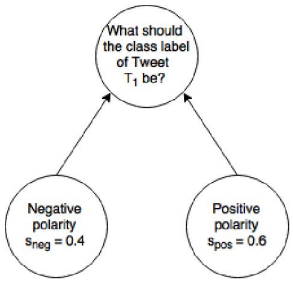

Fig. 2. Example result of the initial classification step shown as a tree.

IV. Polarity classification

In order to classify sentences as positive or negative we follow a two-step procedure:

1) Classify instance using a trained classifier, e.g. Random Forests or Support Vector Machines
2) Argue about classification and change it if we can either find sufficient arguments for a different class label or against the one selected by the classifier

A. (1) Initial classification

In Argumentation terms we treat the possible class labels as answers to the question of what the class label for the instance in question should be, as illustrated by the example shown in figure 2. Each answer has assigned a score and the answer with the higher score is eventually chosen as the class label for the instance in question. We first classify the instance using a model trained on one of the corpora described in section V. To train classifiers and label instances we use a simple *binary Bag-of-Words (BOW)* [30] representation of our instances. The classifiers are then trained using the *Weka* toolbox [31], [32]. This provides us with the initial score for our two possible answers, positive or negative. The score is assigned based on either confidence of the classification (if the classifier provides one), or the classification performance on the training corpus. If, for example, we use a *Naive Bayes'* classifier [33], the class label chosen comes with a confidence value. If we use a non-probabilistic classifier, such as Support Vector Machines, we use the $F1$ measure obtained during training in place of the confidence value. Say we are trying to label the Tweet shown in section III and our classifier wrongly labels it as positive. The classifier provides us some confidence for this classification, say $s_{pos} = 0.6$. We can then view the result of step (1) as a tree as shown in figure 2. In the second step of the classification procedure, this tree is augmented with arguments, changing the score of our two answers by applying the ESAA algorithm, as we describe next.

B. (2) Arguing about classification

Once we have classified an instance as described above we extend the resulting debate tree by identifying arguments that are applicable to the instance in question. The arguments we currently use are made up of sentimental keywords, which, in turn, are taken from a list of positive and negative opinion words developed by Hu, Liu and colleagues (see e.g. [34]). We use a total of $6,815$ arguments, $2,014$ whose conclusion

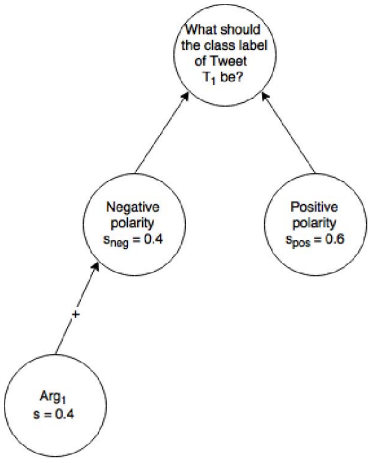

Fig. 3. Example result of the initial classification step, augmented with one argument in support of the *Negative Polarity* answer.

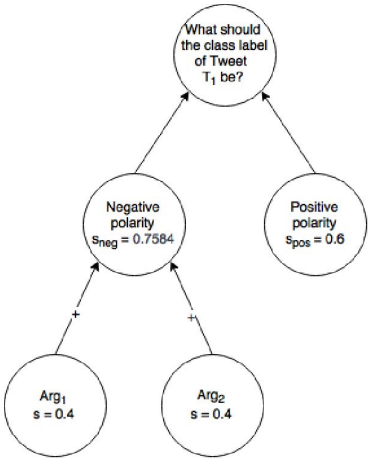

Fig. 4. Example result of the second classification step shown as a tree, augmented with two arguments in support of the *Negative Polarity* answer, and with the answer score recalculated accordingly.

Premise	Conclusion	Score
worthless	neg	0.4
troublesome	neg	0.4
trashy	neg	0.4
stupid	neg	0.4
infuriating	neg	0.4
inconveniently	neg	0.4
furious	neg	0.4
support	pos	0.4
superb	pos	0.4
promising	pos	0.4
impressive	pos	0.4
healthy	pos	0.4
generous	pos	0.4
constructive	pos	0.4

TABLE I. EXAMPLE ARGUMENTS

is *Positive* and $4,801$ whose conclusion is *Negative*. We show some example arguments in table I. Note that all arguments shown are assigned the same score. We are investigating ways of assigning strength values to individual arguments or groups of arguments, but at this time we consider all arguments to be equal. This is not because we believe that they should be, but rather a choice owed to the current lack of ways to make informed choices for scores. We discuss possible ways of making score assignment flexible in section VIII. To illustrate how we incorporate arguments in our classification take again Tweet T_1 and our example argument, Arg_1: Applying Arg_1 to T_1 yields the argument tree shown in figure 3. The arc connecting it to the answer *Negative polarity* is labeled with a plus, indicating that Arg_1 *supports* the answer. It is not a straightforward choice whether Arg_1 should support this answer, as it does here, whether it should attack the *Positive Polarity* answer, or if it should do both. We currently model the majority of our arguments as supports, but we will need to develop ways of modelling certain arguments as attacks, others as supports, and yet others as both. Note that in figure 3 we have not recalculated the score of the answer receiving an argument. In figure 4 we extend our example with further arguments and recalculate the scores. We do so using the ESAA algorithm described in [27]. As we can see, the (correct) answer now has the higher score and we would hence choose Neg as our final label.

Corpus	posCount	negCount	totalCount
Sanders Twitter	570	654	1,058
STS Twitter	588	1,211	1,799
film reviews	5,331	5,331	10,662

TABLE II. CORPORA OVERVIEW WITH COUNTS FOR POSITIVE, NEGATIVE AND TOTAL INSTANCES

V. CORPORA

To determine the merit of our method we have chosen three corpora to experiment with. We show an overview of the corpora in table II. All corpora are annotated for polarity, with every instance being labeled as either *positive* or *negative*. Two of the corpora are comprised of annotated Tweets, while the third corpus is made up of positive and negative sentences taken from film reviews. Using these three corpora gives us the opportunity to investigate classification performance on corpora that differ to lesser or stronger degrees. The Twitter corpora share the style of writing that is particular to the platform, yet the topics which the Tweets are on differ from one corpus to another. The film review corpus, on the other hand, shares neither writing style nor topic with the Twitter corpora. We describe each corpus below.

A. Sanders Twitter corpus

One of the two Twitter corpora we use to conduct our experiments is the *Sanders* corpus described in [35]. The corpus, developed by Sanders Analytics (http://help.sentiment140.com/home), is comprised of $5,513$ manually annotated Tweets. Each Tweet belongs to one of four categories, *Apple*, *Google*, *Microsoft* or *Twitter*. Table III shows a detailed breakdown of how the corpus is made up and table IV shows descriptions of the four classes each category is broken down into. For our purposes, binary polarity classification, we use a subset of the corpus, namely all Tweets that are labeled positive or negative. Some examples of the Tweets selected form the corpus are shown in table V.

Topic	# Positive	# Neutral	# Negative	# Irrelevant
Apple	191	581	377	164
Google	218	604	61	498
Microsoft	93	671	138	513
Twitter	68	647	78	611

TABLE III. CLASS AND CATEGORY DISTRIBUTION OF THE SANDERS TWITTTER CORPUS

Class	Description
Positive	- Positive indicator or topic
Neutral	- Neither positive nor negative indicators - Mixed positive and negative indicators - On topic, but indicator indeterminable - Simple factual statements - Questions with no strong emotions indicated
Negative	- Negative indicator on topic
Irrelevant	- Not English language - Not on topic (e.g. spam)

TABLE IV. DESCRIPTION OF THE SANDERS TWITTER CORPUS CLASSES

B. STS Twitter corpus

The second Twitter corpus we use is taken from the *STS Twitter corpus* [29]. The STS corpus has been developed with a focus on annotating entities in Tweets alongside their sentiment. The corpus is comprised of $2,034$ Tweets, each of which containing a mention of one of the entities shown in table VI. To ensure that the corpus only contains Tweets on topics that are different form those that make up the Sanders corpus we remove all Tweets from the *Technology* category, giving us a total of $1,799$ Tweets in this corpus, of which $1,211$ are negative and the remaining 588 are positive. We show some example Tweets in table VII.

C. Film review corpus

The third corpus we use, which is also the only one not comprised of Tweets, is a corpus made up of sentences taken from film reviews. The corpus was built by Pang and colleagues as part of a larger Sentiment Analysis task of categorising film reviews [11]. It is comprised of $10,662$ sentences, with $5,331$ sentences labeled as positive and the other $5,331$ labeled as negative. All data is taken from the rotten tomatoes website (http://www.rottentomatoes.com). We show some example sentences from the corpus in table VIII. In their research Pang and colleagues address not only determining the polarity of text, but a broader rating-inference problem, in which they attempt to infer start-ratings on a scale from one to five. Though we focus on a binary decision problem here, it would be worthwhile to apply our method to more fine-grained problems such as the inference of ratings.

Class	Sentence
Negative	- Why is #Siri always down @apple - yo @apple this update is a disaster - I hate #Microsoft PowerPoint! - #Microsoft licensing process is annoying !!! - @apple why is my iPhone battery so crappy #fail
Positive	- #Google + #Samsung = Perfect #Icecream sandwich #GalaxyNexus - @Apple: Siri is amazing!!! Im in love! - Great up close & personal event @Apple tonight in Regent St store! - I keep forgettin how much i really like #Twitter lol - #Microsoft store here I come to spend my hard earned cash. #vslive

TABLE V. EXAMPLE TWEETS TAKEN FROM THE SANDERS TWITTER CORPUS

Concept	Most frequent entities	Second-most frequent entities
Person	Taylor Swift, Obama	Oprah, Lebron
Company	Facebook, Youtube	Starbucks McDonalds
City	London, Vegas	Sydney, Seattle
Country	England, US	Brazil Scotland
Organisation	Lakers, Cavs	Nasa, UN
Technology	iPhone, iPod	Xbox, PSP
HealthCondition	Headache, Flu	Cancer, Fever

TABLE VI. DESCRIPTION OF THE STS CORPUS CATEGORIES

Class	Tweet
Negative	- Hayfever time not good! - I'm doing my homework. Its gosh darn hard!! - this week is not going as i had hoped - I'm so tired of work...i need a life.... - I don't understand... I really don't - @ObamaNews Your pages are being redirected to nowhere.
Positive	- @AnnaSaccone Love your new cards! I would definitely hire you - Listening to love story by taylor swift in the car and singing along - Nice my contract was extended for another month - just got home from soccer. Mcdonalds is sooo good - Momz just made it back from Vegas yayyyyy!

TABLE VII. EXAMPLE TWEETS TAKEN FROM THE STS TWITTER CORPUS

Class	Sentence
Negative	- simplistic, silly and tedious. - doesn't add up to much. - constantly slips from the grasp of its maker. - it's mildly amusing , but i certainly can't recommend it. - on its own, its not very interesting . as a remake, its a pale imitation. - this 100-minute movie only has about 25 minutes of decent material.
Positive	- highly engaging. - the entire movie establishes a wonderfully creepy mood. - the film has several strong performances. - it's a satisfying summer blockbuster and worth a look. - a model of what films like this should be like. - clever, brutal and strangely soulful movie.

TABLE VIII. EXAMPLE SENTENCES TAKEN FROM THE FILM REVIEW CORPUS

VI. EVALUATION

We discuss three experiments with which we have evaluated the performance of our method. In each of the experiments we train a classification model on one corpus and classify instances from another. We are provided with a natural split between training and test data. We simply train our models on one corpus and classify *all* instances from another corpus. For each of the three experiments we have trained and compared three different models, *Naive Bayes' (NB)*, *Support Vector Machines (SVM) [36]* and *Random Forests*. To train the Support Vector Machines we use *Radial Basis Function (RBF)* kernels [37], one of the more popular kernels. When training the Random Forests we have chosen a Forest size of 256, which has been shown to be a good tree count with respect to trading off classification performance and computational demand [38]. In each table method (A) denotes the classification without the use of arguments and method (B) that with arguments.

A. Sanders vs. STS

In the first iteration of our experiments we have used the STS Twitter corpus to train the classification model, which we then tested on the Sanders corpus. This run constitutes the only one in which we compare corpora whose contents have been collected from the same source, i.e. Twitter. The difference between the two corpora, in this case, is hence not their source, and thus their style, but rather the topic the

Classifier	Method	Precision	Recall	F1	Accuracy
Naive Bayes'	(A)	.61	.60	.61	.61
	(B)	.66	.66	.66	.66
Random Forest	(A)	.77	.55	.64	.57
	(B)	.67	.63	.65	.64
Support Vector Machines	(A)	.69	.51	.59	.53
	(B)	.66	.61	.63	.63

TABLE IX. CLASSIFICATION PERFORMANCE WITH THE CLASSIFIERS TRAINED ON THE STS CORPUS AND TESTED ON THE SANDERS CORPUS,

Classifier	Method	Precision	Recall	F1	Accuracy
Naive Bayes'	(A)	.51	.50	.51	.50
	(B)	.65	.64	.65	.64
Random Forest	(A)	.56	.51	.53	.51
	(B)	.65	.64	.65	.64
Support Vector Machines	(A)	.51	.50	.51	.50
	(B)	.65	.64	.65	.64

TABLE X. CLASSIFICATION PERFORMANCE WITH THE CLASSIFIERS TRAINED ON THE SANDERS CORPUS AND TESTED ON THE FILM REVIEWS CORPUS

instances are concerned with. As discussed in sections V-A and V-B, while the Tweets that make up the Sanders corpus are concerned with certain technological entities, the STS corpus is made up of Tweets that mention persons, locations, etc. Any overlap in topical domain was avoided by manually pruning the STS corpus for Tweets dealing with the same topics that make up the Sanders corpus. Table IX shows the results using the STS corpus for training and testing on the Sanders corpus.

B. Sanders vs. film reviews

The second run of experiments pits the Sanders Twitter corpus against the film reviews corpus. Again, on the one hand we train a classifier using the Sanders Twitter corpus. Here, however, we classify instances from the film reviews corpus, for which we show the classification outcome in table X. In this run we thus couple two corpora that are not only topically different, as was the case in the previous experiment, but, in addition, are also collected from different sources. Hence we are dealing not just with different contents, but with different styles of writing, different use of colloquial language, etc.

C. STS vs. film reviews

For our third and final experiment we use the STS Twitter corpus and the film reviews corpus. Table XI shows the outcome of training models on the film reviews corpus and classifying instances from the STS Twitter corpus. The setting in this experiment is similar to that described in section VI-B, where both topic and source of the two corpora are different. In this case, however, the film reviews corpus is used for training instead of testing.

Classifier	Method	Precision	Recall	F1	Accuracy
Naive Bayes'	(A)	.59	.59	.59	.64
	(B)	.68	.68	.68	.72
Random Forest	(A)	.61	.58	.59	.67
	(B)	.68	.65	.66	.72
Support Vector Machines	(A)	.62	.59	.6	.68
	(B)	.69	.66	.67	.73

TABLE XI. CLASSIFICATION PERFORMANCE WITH CLASSIFIERS TRAINED ON THE FILM REVIEWS CORPUS AND TESTED ON THE STS CORPUS

VII. DISCUSSION

We first discuss the results of the experiments described in sections VI-A to VI-C. Based on this, in section VII-B we raise some issues for consideration regarding the question of whether it may be useful to distinguish between *near-domain* and *out-of-domain* classification.

A. Experimental results

In section VI we have presented three runs of experiments that perform out-of-domain classification with one of the possible combinations of employing two of three corpora used in this work, one to train a model and another to test it. The results of simply using a classifier are broadly as one may expect. The classification accuracy tends to be rather low, with some outcomes being very close to what we would get from randomly assigning labels to instances. This is particularly true when trying to use one of the Twitter corpora in unison with the film review corpus. When training a model on one of the Twitter corpora and testing on the other, the results somewhat improve. Again, this may be expected, and in section VII-B we discuss whether a distinction between *near-domain* classification, e.g. Twitter plus Twitter, and *out-of-domain* classification, i.e. Twitter plus film reviews, could be useful.

When comparing the results yielded by adding our arguments to the classification procedure we see improved performance across the board. While Naive Bayes' and Support Vector Machines classifiers yield a similar improvements, i.e. $F1$ measures improved by .083 and .082, respectively, using Random Forest classifiers gives us an average improvement of .065 on $F1$ measures across the out-of-domain experiments. When we consider overall average performance of the classifiers, however, we find no notable difference between the classifiers. One may hence argue that using Random Forests may actually be preferable over the other two classifiers, since the lower achieved improvement is owed to the baseline classification performance of the Random Forest being better than that of the other two classifiers. On the other hand the Naive Bayes' classifier performs slightly, though not significantly, better than the other two classifiers. Whether we simply want to choose the classifier that yields the best overall performance or the one that also performs as well as possible on the baseline remains an open question that requires further addressing. Additionally we may need to conduct further experiments with other classifiers. We summarise the average classification performance measures in table XII. The averages include experiments in which we reverse the roles of the corpora to those shown in this paper. Due to space limitations we have not shown these results, individually, but the are broadly in line with what we encounter in the results reported here. Overall it appears that adding our arguments helps us more with improving recall than it does with improving precision. The average increase in recall lies at .093, while the average increase in precision is .05. Again, however, the higher increase is owed to a lower baseline. It seems the classifiers, on their own, struggle more with achieving reasonable recall than they do with precision. In fact, the average precision, at .66 is higher than the average recall at .643.

Classifier	Method	Precision	Recall	F1	Accuracy
Naive Bayes'	(A)	.58	.57	.58	.59
	(B)	.66	.66	.66	.66
Random Forest	(A)	.64	.55	.59	.59
	(B)	.66	.63	.65	.67
Support Vector Machines	(A)	.61	.53	.57	.58
	(B)	.66	.64	.65	.67

TABLE XII. OVERVIEW OF AVERAGE CLASSIFICATION PERFORMANCE OVER ALL CORPUS COMBINATIONS

Method	Twitter + Twitter	Twitter + film reviews
(A)	.61	.56
(B)	.65	.65

TABLE XIII. SUMMARY OF AVERAGE $F1$ SCORES, SPLIT ACCORDING TO WHETHER BOTH TRAINING AND TEST DATA WERE TWITTER CORPORA OR ONE OF THE TWO WAS THE FILM REVIEW CORPUS

B. Near-domain vs. out-of-domain

One may expect to observe better classification results when classifying Twitter data with a model trained on other Twitter data when compared to training models on the film review corpus and classifying Twitter data, or the other way around. This is indeed the case when we consider the average $F1$ scores achieved when using the classifiers, only, as shown in table XIII. When, however, we consider our arguments, as well, we achieve nearly the exact same $F1$ measures for either combination of corpora. We believe that this may occur for one, or both, of two reasons:

1) The arguments we have chosen are more suitable to the film review corpus and we hence encounter more applicable arguments that argue for the true class of instances in this corpus.
2) The increased rate of errors when combining one of the Twitter corpora with the film review corpus simply yields more instances in which the arguments stand in contradiction to the class label suggested by the classifier

Based on the results of our experiments we believe that the difference between what we call *Near-domain* classification and *Out-of-domain classification* may provide useful guidance. In our case, the fact that the style of writing of the contents of two of the corpora is the same may help our classification performance, even though the topics discussed are considerably different. This may hold lessons for choosing future training data when considering out-of-domain classification. There may be corpora that are *nearer* to the problem at hand than others, whether that be in terms of content or style. Choosing corpora for classification that are as near to our domain as possible should help us improve out-of-domain classification, in general.

VIII. CONCLUSION

In this paper we have presented our work on developing solutions for out-of-domain sentiment polarity classification using Argumentation formalisms. We have used three corpora, two comprised of Tweets and one of sentences taken from film reviews, to train classification models on each and use those models to classify instances from the other corpora. As expected, large parts of our experiments yielded rather poor results when simply using a model trained on one corpus to classify instances from another. When using the described Argumentation formalisms to add largely domain-independent knowledge to our classification procedure, however, we find that classification performance improves in all scenarios tested. To conclude our paper we raise some issues for consideration that we have yet to address in our research.

A. Other arguments

At this juncture we have focused on using keywords that have a mostly unambiguous polarity to build our arguments. There may, however, be other sources from which we can construct arguments, which may either complement or replace our current set of arguments. Such other arguments may formalise knowledge about text, such as that described in section VIII-B In parallel to introducing new arguments we will need to experiment with various subsets of arguments, as well as score settings, which we discuss in section VIII-C. It may well be that certain arguments actually turn out to hinder classification. In that case we would need to either remove such arguments, change their score, or reconsider the conclusion we draw from the premise of an argument.

B. Other representations

To train classifiers we have represented our instances using a Bag-of-Words approach. We have yet to delve into other possible representations, which may help us improve the baseline classification, and, depending on the subsequent impact of our arguments, the final classification, as well. Numerous feature-based representations have been developed for sentiment polarity classification. Some examples include the following:

- Structural features: Consider e.g. the parse of a phrase or sentence to deduce insights into their sentimentality [3]
- Knowledge based features: Use external knowledge bases, such as Wordnet [39] to identify relevant relations between words, etc. [40], [13]
- Modification features: For word polarity classification, take into account the type of words that precede or follow, and thus potentially modify, the word in question [3]
- Document features: Gather features from (meta) data about the document from which the text snippet in question is taken [3]

Building feature-based representations of instances may be especially helpful in our scenario. It may allow us to somewhat lessen the domain-dependence from the get-go, as Bag-of-Words representations are prone to suffer from it more than more generic, feature-based, representations.

C. Argument scores

In table I we have shown example arguments, each of which has assigned to it a score. We have noted that, at this juncture, each argument has the same score. By changing the scores for all arguments at a time we can change the overall impact arguments can have on reconsidering the class label suggested by the classifier. If we increase the argument scores we need fewer arguments to overturn a suggested

class label, if we decrease the scores we need to find more applicable arguments, instead. It may, however, be useful to assign different scores to either individual arguments or groups of arguments. This would allow us to tailor the impact an argument has to how reliably its conclusion holds, for example. Take the argument *superb pos 0.4*. If, for instance, we found that instances containing the word superb (possibly corrected for negations) are almost exclusively positive, while those containing other keywords are not as reliable, we may want to increase the score of this argument, and others that are similarly reliable, but leave the score of other arguments unchanged.

REFERENCES

[1] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and trends in information retrieval*, vol. 2, no. 1-2, pp. 1–135, 2008.

[2] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining." in *LREC*, vol. 10, 2010, pp. 1320–1326.

[3] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," in *Proceedings of the conference on human language technology and empirical methods in natural language processing*. Association for Computational Linguistics, 2005, pp. 347–354.

[4] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[5] B. Liu, "Sentiment analysis and opinion mining," *Synthesis Lectures on Human Language Technologies*, vol. 5, no. 1, pp. 1–167, 2012.

[6] B. Liu and L. Zhang, "A survey of opinion mining and sentiment analysis," in *Mining text data*. Springer, 2012, pp. 415–463.

[7] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language," *Language resources and evaluation*, vol. 39, no. 2-3, pp. 165–210, 2005.

[8] E. Riloff and J. Wiebe, "Learning extraction patterns for subjective expressions," in *Proceedings of the 2003 conference on Empirical methods in natural language processing*. Association for Computational Linguistics, 2003, pp. 105–112.

[9] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, 2002, pp. 79–86.

[10] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2004, p. 271.

[11] ——, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2005, pp. 115–124.

[12] L. Chen, W. Wang, M. Nagarajan, S. Wang, and A. P. Sheth, "Extracting diverse sentiment expressions with target-dependent polarity from twitter." in *ICWSM*, 2012.

[13] A. Montejo-Ráez, E. Martínez-Cámara, M. T. Martín-Valdivia, and L. A. Ureña-López, "Ranked wordnet graph for sentiment polarity classification in twitter," *Computer Speech & Language*, vol. 28, no. 1, pp. 93–107, 2014.

[14] D. Vilares, M. A. Alonso, and C. Gómez-Rodríguez, "On the usefulness of lexical and syntactic processing in polarity classification of twitter messages," *Journal of the Association for Information Science and Technology*, 2015.

[15] J. Villena-Román, S. Lana-Serrano, C. Moreno, J. García-Morera, and J. C. G. Cristóbal, "Daedalus at replab 2012: Polarity classification and filtering on twitter data." in *CLEF (Online Working Notes/Labs/Workshop)*, vol. 60, 2012.

[16] R. Xia, C. Zong, X. Hu, and E. Cambria, "Feature ensemble plus sample selection: domain adaptation for sentiment classification," *Intelligent Systems, IEEE*, vol. 28, no. 3, pp. 10–18, 2013.

[17] T. Li, V. Sindhwani, C. Ding, and Y. Zhang, "Knowledge transformation for cross-domain sentiment classification," in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2009, pp. 716–717.

[18] D. Bollegala, D. Weir, and J. Carroll, "Cross-domain sentiment classification using a sentiment sensitive thesaurus," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 25, no. 8, pp. 1719–1731, 2013.

[19] A. Weichselbraun, S. Gindl, and A. Scharl, "Extracting and grounding context-aware sentiment lexicons," *IEEE Intelligent Systems*, vol. 28, no. 2, pp. 39–46, 2013.

[20] ——, "Enriching semantic knowledge bases for opinion mining in big data applications," *Knowledge-Based Systems*, vol. 69, pp. 78–85, 2014.

[21] A. C.-R. Tsai, C.-E. Wu, R. T.-H. Tsai, and J. Y.-j. Hsu, "Building a concept-level sentiment dictionary based on commonsense knowledge," *IEEE Intelligent Systems*, no. 2, pp. 22–30, 2013.

[22] S. J. Pan, X. Ni, J.-T. Sun, Q. Yang, and Z. Chen, "Cross-domain sentiment classification via spectral feature alignment," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 751–760.

[23] I. Rahwan and G. R. Simari, Eds., *Argumentation in AI*. Springer, 2009.

[24] P. M. Dung, "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games," *Artificial intelligence*, vol. 77, no. 2, pp. 321–357, 1995.

[25] C. Cayrol and M. Lagasquie-Schiex, "On the acceptability of arguments in bipolar argumentation frameworks," in *Proc. ECSQARU*, 2005, pp. 378–389.

[26] P. Baroni, M. Romano, F. Toni, M. Aurisicchio, and G. Bertanza, "Automatic evaluation of design alternatives with quantitative argumentation," *Arg. & Comp.*, vol. 6, no. 1, pp. 24–49, 2015.

[27] V. Evripidou and F. Toni, "Quaestio-it.com – A social intelligent debating platform," *Journal of Decision Systems*, vol. 23, no. 3, pp. 333–349, 2014.

[28] J. Leite and J. Martins, "Social abstract argumentation," in *Proc. IJCAI*, 2011.

[29] H. Saif, M. Fernandez, Y. He, and H. Alani, "Evaluation datasets for twitter sentiment analysis: a survey and a new dataset, the sts-gold," 2013.

[30] G. Salton and M. J. McGill, "Introduction to modern information retrieval," 1986.

[31] S. R. Garner *et al.*, "Weka: The waikato environment for knowledge analysis," in *Proceedings of the New Zealand computer science research students conference*. Citeseer, 1995, pp. 57–64.

[32] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *ACM SIGKDD explorations newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[33] G. H. John and P. Langley, "Estimating continuous distributions in bayesian classifiers," in *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1995, pp. 338–345.

[34] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2004, pp. 168–177.

[35] N. J. Sanders, "Sanders-twitter sentiment corpus," *Sanders Analytics LLC*, 2011.

[36] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[37] M. D. Buhmann, *Radial basis functions: theory and implementations*. Cambridge university press, 2003, vol. 12.

[38] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas, "How many trees in a random forest?" in *MLDM*. Springer, 2012, pp. 154–168.

[39] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[40] S.-M. Kim and E. Hovy, "Determining the sentiment of opinions," in *Proceedings of the 20th international conference on Computational Linguistics*. Association for Computational Linguistics, 2004, p. 1367.