ADAPTATION OF SVM FOR MIL FOR INFERRING THE POLARITY OF MOVIES AND MOVIE REVIEWS

Joana Correia[1,2,3], *Isabel Trancoso*[2,3], *Bhiksha Raj*[1]

[1]Language Technologies Institute, Carnegie Mellon University, USA
[2]Instituto Superior Tecnico, Portugal
[3]INESC-ID, Portugal

ABSTRACT

Polarity detection is a research topic of major interest, with many applications including detecting the polarity of product reviews. However, in some cases, the polarity of the product reviews might not be available while the polarity of the product itself might be, prohibiting the use of any form of fully supervised learning technique. This scenario, while different, is close to that of multiple instance learning (MIL). In this work we propose two new adaptations of support vector machines (SVM) for MIL, θ-MIL, to suit this new scenario, and infer the polarity of products and product reviews. We perform experiments on the proposed methods using the IMDb movie review corpus, and compare the performance of the proposed methods to the traditional SVM for MIL approach. Although we make weaker assumptions about the data, the proposed methods achieve a comparable performance to the SVM for MIL in accurately detecting the polarity of movies and movie reviews.

Index Terms— sentiment analysis, Doc2Vec, multiple instance learning, SVM, IMDb

1. INTRODUCTION

Polarity detection in documents, meaning inferring if the overall sentiment of a piece of text is positive or negative, has attracted increasing attention in recent years [1], particularly with the advent of online reviews for products or media content, and the increase in popularity of online platforms like Twitter [2]. With the large number of reviews available online, polarity detection could provide a valuable tool to benefit both the audience of a product with a succinct summary of the sentiment for that product, as well as for their makers, with a succinct feedback from the users of the product.

Movie reviews are a prime example of documents where people express their sentiment for something specific. Moreover, some reviews may be accompanied by a quantified measure of polarity, like a rating based on a 5 star system, that can be used to compute the polarity after thresholding. In such scenario, where the movie reviews and the polarities are available, it is possible to solve the problem of inferring the polarity of a new review using a number of supervised machine learning approaches. However, instance-level ratings might not be always available, while movie-level polarities might. This scenario resembles, to some extent, that of the multiple instance learning (MIL), where the data is assumed to have some ambiguity in how labels are assigned. Particularly, rather than having labels associated to data instances, they are assigned to *sets* or *bags* of instances. In the MIL paradigm the main assumption is that every bag associated with a positive label contains at least one instance that is positive, while a bag associated with a negative label contains strictly negative instances. The task is to predict the label of a new bag and of its instances [3].

However, the main MIL assumption of the contents of the bags is not adequate for scenarios of products and content review where only the overall product rating is known. A movie that is considered bad is unlikely to have only negative reviews, while a good movie is unlikely to be considered good merely because of having at least one good review. A more reasonable assumption would be that the reviews of a good movie are mostly good and the reviews of a bad movie are mostly bad.

In this work, we discard the MIL assumption about the relationship between bag and instance labels, and adopt the weaker assumption that the bag labels are determined by the dominant label in the instances of the bag. We propose two solutions for this new problem by introducing two adaptations of the formulations of the MIL as a maximum margin problem, using support vector machines (SVMs), that take into account the new, weaker assumptions, to which we refer to as θ-MIL. The proposed formulations are mixed-integer quadratic problems, which can be solved heuristically. We apply our algorithms to a movie review corpus and use them to infer the polarity of both movies and movie reviews. To extract features from the raw movie reviews we use Doc2Vec, an unsupervised framework that learns a continuous distributed representation for text documents of variable length, and where

This work was partially supported by the grant SFRH/BD/103402/2014 from the Portuguese Foundation for Science and Technology

the feature space is semantically meaningful, i. e. the semantic similarity between two words can be computed from the similarity between their respective feature vectors.

We believe that making the assumption that the bag labels are determined by the dominant instance label in that bag, rather than the typical MIL assumption, is a more reasonable assumption for a number of scenarios related to products and media content reviews. In them, the sentiment for the product or media content is likely related to the average sentiment across all reviews. So the applications for our proposed approach are numerous.

This paper is organized as follows: In Section 2 we review related work in polarity detection, and MIL; in Sections 3 and 4 we introduce doc2vec and SVM for MIL, the two main techniques we used in this work; in Section 5 we introduce the main contribution of this work, the proposed methods θ-MIL for a scenario of products review; in Section 6 we perform some experiments comparing our proposed methods to the existing MIL for SVM; and finally, in Section 7 we draw some conclusions.

2. RELATED WORK

Our work ties two topics: polarity detection for documents, and multiple-instance learning of classifiers from "weak" labels.

Polarity detection in documents is a form of sentiment analysis that focuses on detecting whether the document expresses a positive or negative opinion in general or about a given entity (e.g. a product, a person, a political issue, etc). It has attracted increasing attention in recent years, following the work in [4] and [5] who employed a variety of classifiers on word-level features extracted from documents – they found that SVMs generally yield better performances than other classifiers. Current state-of-the art approaches to sentiment analysis try to go beyond word-level features, like in [6], that introduced Recursive Neural Tensor Network for sentence parsing, or [7] that introduced distributed representations for documents, as an extension for word2vec.

Multiple-instance learning (MIL) [3] is a generalization of supervised classification in which training class labels are associated with *sets*, or *bags*, of instances instead of individual instances. The true label of every instance remains hidden and is only indirectly inferred through the labels associated to the bags. The main assumption of MIL is that negative bags only contain negative instances, while positive bags contain at least one positive instance. Typical applications of MIL are drug design [3], image indexing from content-based image retrieval [8], or text categorization [9].

3. DOC2VEC

The first efforts to obtain vector representations for words using neural networks began in the last decade [10] [11] [12] [13] [14]. The main idea of these works was that each word could be represented by a vector, such that simple arithmetic operations between vectors could be used to predict words in the context. Typically, this means that the vector representations of words live in a space where the distance between vector representations correlates to the semantic similarity of their corresponding words. For instance, the word representation of "big" should have a smaller distance to the word representation of "large" than to the one of "dog".

A natural extension of these successful techniques is to go beyond word level representations and try to achieve phrase- or sentence-level representations [15] [16] [17] [18] [19]. One simple approach to achieve a sentence representation, for instance, is to perform a weighted average of all the word representations in that sentence. Naturally, the order of the words is lost and with it, part of the information from the sentence.

Doc2Vec, or Paragraph Vector [7], emerged as an unsupervised framework that learns a continuous distributed representation for text documents of variable length. From a single sentence to several paragraphs, Doc2Vec can preserve some information related to word ordering. In this framework, particularly in the distributed memory model, every document is mapped to a unique document vector and every word in the vocabulary is mapped to a unique word vector.

Given a sequence of training words, the goal of the distributed memory model is to maximize the average log probability of a word given its context. Then the prediction task is achieved via a multiclass classifier.

4. SVM FOR MIL

One of the most usual scenarios in pattern recognition problems is the fully supervised one. It occurs when the available training set is made of labeled instances, $(\mathbf{x}_i, y_i) \in \mathbb{R}^d \to \mathcal{Y}$, generated independently from an unknown distribution. If the labels are binary then, $\mathcal{Y} = \{-1, 1\}$. In the MIL scenario, this problem is generalized by the ambiguity in the labeling of the instances. Instances, $\mathbf{x}_1, ..., \mathbf{x}_n$, are grouped into *bags*, $\mathbf{B_1}, ..., \mathbf{B_m}$, with $\mathbf{B}_I = \{\mathbf{x}_i : i \in I\}$, for non-overlapping $I \subseteq \{1, ..., n\}$. Each bag, $\mathbf{B}_I$, is associated to a label, $\mathbf{Y}_I$; if $\mathbf{Y}_I = 1$, then there is at least one bag instance, $\mathbf{x}_i \in \mathbf{B}_I$ with $y_i = 1$, or if $\mathbf{Y}_I = -1$, then all $\mathbf{x}_i \in \mathbf{B}_I$ have $y_i = -1$.

There are multiple solutions to MIL problems, and, among other approaches, MIL can be formulated as a maximum margin problem, and be solved by extensions of SVMs [9]. In [9], the authors propose two such approaches: the first treats the instance labels as unobserved integer variables, subject to the constraints defined by the positive bag labels; the second generalizes the notion of a margin from instances to bags and aims to maximize the bag margin.

In more detail, the first approach, *mi*-SVM, can have its mixed integer formulation of MIL as a generalized soft-margin SVM, and its primal form can be written as follows:

$$
\begin{aligned}
\min_{\{y_i\}} \min_{\mathbf{w},b,\xi} \quad & \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i \\
\text{s.t.} \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1 - \xi_i \\
& \xi_i \geq 0 \\
& y_i \in \{-1, 1\} \\
& \sum_{i \in I} \frac{1 + y_i}{2} \geq 1, \forall I \text{ s.t. } Y_I = 1 \\
& y_i = -1 \forall I \text{ s.t. } Y_I = -1
\end{aligned} \tag{1}
$$

where the optimization variables $\mathbf{w}$, b, ξ_i, and y_i, are, respectively, the weight vector, a scalar, a scalar, and the predicted instance label for example i. Y_I is the bag label for bag I.

The second approach, *MI*-SVM, is formulated as a quadratic mixed integer problem, as follows:

$$
\begin{aligned}
& \min_s \min_{\mathbf{w},b,\xi} \frac{1}{2}||\mathbf{w}||^2 + C\sum_I \xi_I \\
& \text{s.t.} \\
& \forall I : Y_I = 1 \wedge (-\langle \mathbf{w}, \mathbf{x}_i \rangle - b) \geq 1 - \xi_I \\
& \text{or } Y_I = 1 \wedge (\langle \mathbf{w}, \mathbf{x}_{\mathbf{s}(I)} \rangle + b) \geq 1 - \xi_I \\
& \xi_I \geq 0
\end{aligned} \tag{2}
$$

where the optimization variables $\mathbf{w}$, b, ξ, and s, are, respectively, the weight vector, a scalar, a scalar, and the instance selector. Y_I is the bag label for bag I. Note that $s(I)$ acts as a selector among the instances of a bag. It will be active for one instance in each positive bag.

In this case, the positive bag margin is defined by the margin of the "most positive" instance.

The difference between the two approaches is essentially that in MI-SVM, the negative instances of the positive bags are ignored, and at the same time, only one instance per positive bag contributes to the optimization problem. On the other hand, in mi-SVM, negative instances in positive bags, as well as one or more positive instances from a positive bag can be support vectors.

5. θ-MIL

In the scope of this work, which is to infer the polarity of movies and movie reviews, the MIL assumption that positive bags have at least one positive instance and negative bags have exclusively negative instances, is not adequate. An alternative for the assumptions regarding bag and instance organization would be to associate to each bag, $\mathbf{B}_I$, a label $\mathbf{Y}_I$, where $\mathbf{Y}_I = sign(\frac{\sum_i (y_i : i \in I)}{|I|})$.

After defining bag labels from the dominating bag label, the problem leaves the scope of MIL and the solutions presented in Section 4 are not valid anymore. However, the problem is still similar to the MIL problem, so the solutions for MIL can be modified to suit this new scenario.

Given a set of bags $\mathbf{B}_I$, their labels $\mathbf{Y}_I$, and the instances of each bag, $\{\mathbf{x}_i : i \in I\}$, the optimal class separating hyperplane with parameters $\mathbf{w}$ and b, and instance labels $\{\mathbf{y}_i : i \in I\}$, can be found by minimizing the same objective as Problem 1, subject to two new constraints: $\sum_{i \in I} \frac{1+y_i}{2|I|} \geq \theta_+, \forall I s.t. Y_I = 1$, and $\sum_{i \in I} \frac{1+y_i}{2|I|} < \theta_-, \forall I s.t. Y_I = -1$. Thus, the adaptation of the mi-SVM, to which we refer to as θ-mi-SVM, can be written as:

$$
\begin{aligned}
\min_{\{y_i\}} \min_{\mathbf{w},b,\xi} \quad & \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i \\
\text{s.t.} \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1 - \xi_i \\
& \xi_i \geq 0 \\
& y_i \in \{-1, 1\} \\
& \sum_{i \in I} \frac{1 + y_i}{2|I|} \geq \theta_+, \forall I \text{ s.t. } Y_I = 1 \\
& \sum_{i \in I} \frac{1 + y_i}{2|I|} < \theta_-, \forall I \text{ s.t. } Y_I = -1
\end{aligned} \tag{3}
$$

We note that the problem remains mixed integer, such as the Problem 1. The first to third constraints remain the same. With this new formulation, we will have at least the fraction θ_+ of the instances of each bag labeled positive in the positive halfspace, and at least the fraction θ_- of the instances of a negative bag in the negative halfspace. At the same time, the margin is maximized with respect to the complete dataset, according to the instance labels that were assigned.

The resulting mixed integer Problem 3 cannot be easily solved. So we employ the following heuristic:

Algorithm 1 θ-mi-SVM optimization heuristics

Input: $\mathbf{x}_i$, B_I, $\mathbf{y}_{B_I}$
Initialize $y_i = Y_I$ for $i \in I$
while *labels change from previous iteration* **do**
  Compute SVM solution $\mathbf{w}, b$ for the train instances and labels
  Compute outputs $f_i = \langle \mathbf{w}, \mathbf{x}_i \rangle + b$ for all $\mathbf{x}_i$ in all bags
  Update $y_i = sgn(f_i)$ for ever $i \in I$
  for *every positive bag* **do**
    if $\frac{\sum_{i \in I} 1+y_i}{2|I|} \geq \theta_+$ **then**
      compute $\mathbf{i}^* = \arg\max_{i, \theta_+ |I|} f_i$
      set $y_{\mathbf{i}^*} = 1$
    end
  end
  for *every negative bag* **do**
    if $\frac{\sum_{i \in I} 1+y_i}{2|I|} < \theta_-$ **then**
      compute $\mathbf{i}^* = \arg\min_{i, \theta_- |I|} f_i$ set $y_{\mathbf{i}^*} = -1$
    end
  end
end
Output $\mathbf{w}, b$

In the above algorithm we use the notation $\arg\max_{i,K} f_i$ to represent the set of indexes of the K highest valued f_i.

The heuristic to solve the Problem 3 involves alternating between two steps. In the first, given the instance labels, we solve the SVM and find the optimal separating hyperplane. In the second, for a given hyperplane, we update the instance labels in order to respect the constraints that at least a fraction θ_+ or θ_- of the instances of positive or negative bags, respectively, will have the same label as their respective bag. Note that although we do not update labels within bags that fail the "if" clause in the algorithm, the instance labels retain their last known state and the number of positives/ negatives in each bag remains such that the original constraint is not violated.

Secondly, we propose an adaptation of the MI-SVM, to which we refer to as θ-MI-SVM. In this case, the goal is to extend the notion of a margin from instance level to the bag level. As such, we define the functional margin of a bag with respect to only the instances with the same predicted label as the bag. So for the positive margin, the optimization problem uses the "most positive" instances and for the negative margin it uses the "most negative" instances, such that each positive and negative bag have at least a fraction θ_+ and θ_-, of the instances being selected as key witnesses, respectively. The new optimization problem can be written as follows:

$$\min_{s} \min_{\mathbf{w},b,\xi} \frac{1}{2}||\mathbf{w}||^2 + C\sum_I \xi_I$$

s.t.

$$\forall I : Y_I = 1 \wedge (\langle \mathbf{w}, \mathbf{x}_{\mathbf{s}(I)} \rangle + b) \geq 1 - \xi_i \wedge |s(I)| \geq \theta_+ |I|$$

$$\text{or } Y_I = -1 \wedge (-\langle \mathbf{w}, \mathbf{x}_{\mathbf{s}(I)} \rangle - b) \geq 1 - \xi_i \wedge |s(I)| \leq \theta_- |I|$$

$$\xi_i \geq 0 \tag{4}$$

where the optimization variables w, b, ξ, and s are the weight vector, a scalar, a scalar, and the instance selector respectively. Note that $s(I)$ selects a fraction of the instances of the bag, an not just one, unlike in Problem 2. More specifically it selects $\theta_+|I|$ or $\theta_-|I|$ instances in positive or negative bags, respectively. Furthermore, since the margins are defined by the instances which have a label that matches their respective bag, this approach ignores the remaining instances within each bag. They are not contemplated in the optimization problem, contrary to the case of θ-mi-SVM.

There can be many initialization of the labels, however [9] recommends initializing the instance labels with the corresponding bag label.

This new problem, as in the MI-SVM formulation, is a mixed integer problem, without an easy solution. Therefore, we use the heuristic shown in Algorithm 2 to solve it iteratively as shown in Algorithm 2.

Algorithm 2 θ-MI-SVM optimization heuristics

Input: x_i, B_I, y_{B_I}
Initialize $\mathrm{x}_I = \sum_{i \in I} \frac{\mathbf{x}_i}{|I|}$ for every bag
Initialize selector variables $s(I)$, where $\sum_{i \in I} s(i) \geq \frac{|I|}{2}$
while *s(I) changes from previous iteration* **do**
  Compute QP solution $\mathbf{w}, b$ for the train instances and labels
  Compute outputs $f_i = \langle \mathbf{w}, \mathbf{x}_i \rangle + b$ for all $\mathbf{x}_i$ in all bags
  for *every positive Bag* **do**
    set $\mathbf{x} = \mathbf{x}_{s(I)}$, where $s(I) = \arg\max_{i,K} f_i$
  end
  for *every negative Bag* **do**
    set $\mathbf{x} = \mathbf{x}_{s(I)}$, where $s(I) = \arg\min_{i,K} f_i$
  end
end
Output: $\mathbf{w}, b$

Similarly to the heuristic of Algorithm 1 for θ-mi-SVM, Algorithm 2 also alternates between two steps: One is, for the given selected instances of every bag, compute the quadratic problem solution and find the optimal separating hyperplane; The other is, given a separating hyperplane, update the selected instances according to the problem constraints. Once the selected variables do not change from the previous iteration, the algorithm ends. We note that, unlike in the MI-SVM algorithm, in this case the instance selector, s, will select one or more instance of a bag, such that at least a fraction θ_+ or θ_- of the instances in the bag are positive or negative, for a positive or negative bag, respectively.

The initialization of the instances can be the bags centroids, as suggested in [9].

6. EXPERIMENTS AND RESULTS

6.1. Corpus

The polarity dataset is a corpus of movie reviews retrieved from the Internet Movie Database (IMDb) archive [5]. The corpus contains 2000 movie reviews in English, where each review is associated to the movie it refers to, and to a rating expressed by the reviewer in stars or some numerical value. The typical review content is a small text where people summarize the story of the movie and highlight the positive or negative aspects that struck them most.

The movie reviews are determined as positive or negative from their ratting as follows: 1) For ratings specified in 5 star systems, 3.5 stars or more is considered positive, 2 stars or less is considered negative; 2) For 4 stars systems, a rating of 3 stars or more is considered positive, 1.4 or less is considered negative; 3) For letter grade systems, B or above is considered positive, C- or below is considered negative.

The polarity dataset contains 1000 positive reviews, 1000 negative reviews, and no neutral reviews. The 2000 reviews

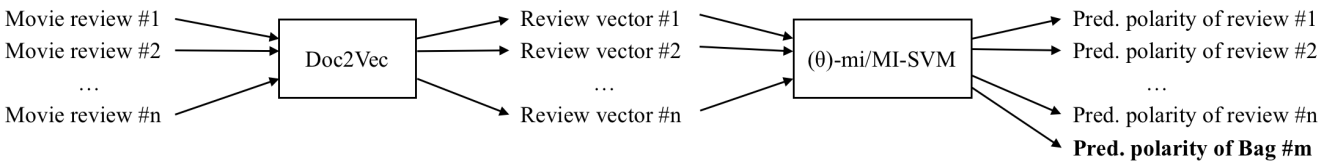

Fig. 1. Example of the proposed framework at test time to predict the polarity of one bag and its instances

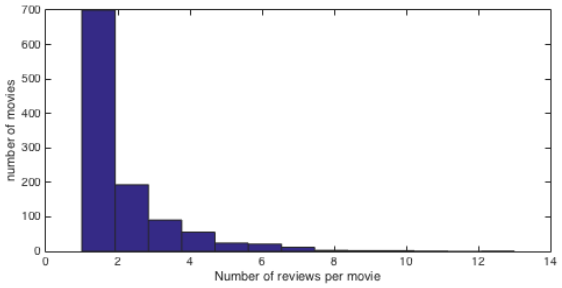

Fig. 2. Histogram of the bag size for the IMDb corpus

cover 1106 different movies. There is at least 1 review per movie, and at most 13 reviews per movie. A distribution of number of reviews per movie can be found in Figure 2.

There are two types of movie-level labels available: The MIL labels, attributed according to the MIL assumption that bags with only negative reviews have a negative bag-level label, and if there is at least one review then the bag-level label is positive; and the majority labels, attributed to a bag according to which is the most frequent instance label for that bag.

6.2. Inferring polarity of movies and movie reviews with MIL

The first experiments we performed aimed at establishing a baseline to which we can compare our proposed methods.

More specifically, we adopt the IMDb movie corpus described in Section 6.1, of 2000 movie reviews, for 1106 different movies, and compute each review's feature vector using Doc2Vec. In this scenario the bags correspond to sets of movie reviews for a given movie. For simplicity we will address them as movies, and their polarity as movie polarity. Then, we split the dataset in two, leaving 830 bags (with 1445 instances) for training, and 276 bags (with 555 instances) for testing the models, where the bags are labeled according to the MIL assumption: negative if all instance are negative, and positive if at least one instance is positive.

The training instances are used to train three classifiers: a fully supervised SVM, a mi-SVM, and a MI-SVM, as described in Section 4. At test time, the remaining test instances are used to compute the performance of the three models. The SVM will predict the polarity of the test reviews, while the mi-SVM and MI-SVM will predict both the polarity of the test reviews and their respective bags. Figure 1 summarizes the proposed framework for the baseline system at test time, for a given bag and its instances, i. e., the reviews of a given movie. The kernel function for the SVMs of the three systems is the radial basis function (RBF). In all of the experiments θ_+ and θ_- were set to 0.5.

Table 1. Performance in accuracy of the three SVM based systems using MIL bag labels for the train and test dataset

MIL bags	**SVM [acc.]**	**mi-SVM [acc.]**		**MI-SVM [acc.]**	
	inst.	*bags*	*inst.*	*bags*	*inst.*
train	95.85	-	89.12	-	89.20
test	86.85	83.03	84.30	82.31	85.74

Table 1. summarizes the classification accuracy of the traained classifiers. Recall that we only have bag-level labels for the training data, and the MIL training framework also learns instance-level labels for each of the bags. The effectiveness of the training algorithm also reflects on the accuracy with which the labeles of instances in the *training* are learnt. Hence, we report both, the accuracy of instance-level label assignment on the *training* set, as well as accuracy on the test set. From there, we note that the best performance was obtained on the test set from the fully supervised SVM, that achieved an accuracy for instance label prediction of 86.8%. This system's purpose is to serve as a comparison between the performance of a fully supervised model and a MIL model. The mi-MIL and MI-MIL performances at instance level were quite similar to the fully supervised SVM, achieving an accuracy on predicting the instance label of 84.3% and 85.7%, slightly lower than the respective respective results from the train set. As for the prediction of the bag-level labels, the systems achieved 83.0% and 82.3% accuracy on the test set.

The similar performance between the fully supervised SVM and the mi-SVM and MI-SVM is likely to be caused, to some extent, by the size of the bags. Since there are many small bags, there is less ambiguity in the data, particularly in bags of size one where the instance label is explicitly known from the MIL assumption.

6.3. Inferring polarity of movies and movie reviews with θ-MIL

Our second experiment aimed at establishing the performance of the methods proposed in this work.

Following the experiments described in section 6.2, we adopt the same IMDb corpus, described in section 6.1, and split it in the same subsets as described in section 6.2: 830 bags (with 1445 instances) for training, and 276 bags (with 555 instances) for testing. This time the bag labels are attributed according to the dominant label in the instances of each bag, in order to match the intuition for the polarity of a movie and its reviews. In this case, the data has more ambiguity than in the traditional MIL scenario, since in this case, there is ambiguity in the instance labels for both the positive and the negative bags, instead of just the positive ones.

Nevertheless, the whole process, from feature extraction to classification, is identical to the one described Section 6.2, except that the mi-SVM and MI-SVM solutions are replaced by θ-mi-SVM and θ-MI-SVM, respectively, for the scenario where the bags are labeled according to the dominating label in their instances, as described in Section 5. Once again, we also train a fully supervised SVM for comparison. An example of the test stage for a given movie and its reviews is shown in Figure 1.

The performance of the three systems against the train and test sets is measured in accuracy and is summarized in Table 2. We can see that the performance of the fully supervised SVM for the train and test sets remains the same as the one reported in Section 6.2, because although the bag labels changed the instance labels remained the same, and those were the only ones used by the fully supervised SVM. Furthermore, the performance of the θ-mi-SVM and θ-MI-SVM with respect to the accuracy in predicting the instance labels on the test set was 76.9% and 82.9%, respectively. The comparatively poorer performance of θ-mi-SVM method to the remaining methods might be related with the inner workings of the method itself: for each bag, the method selects a fraction of the instances to attribute the bag label to, while the remaining fraction is attributed the opposite label. However, there might be a misclassification of the later set. The alternative, as happens in the θ-MI-SVM method, is to discard the later fraction when estimating the model.

Comparing the drop in performance from the train to the test set, we can see that in this case the difference is slightly larger than the results from Table 1, which may mean that the model slightly overfit. The performance of the same methods regarding the accuracy in predicting the bag labels was 82.6% and 83.3% for the test set, a similar performance to that reported in Table 1.

Finally, since the IMDb corpus has bags with different sizes, and many of the bags are of size one, it would be interesting to reevaluate the systems for filtered versions of the corpus, where the smaller bags are discarded. However, we note that since the corpus is small, the subsets of the corpus with large enough bags would become too small to train a robust model.

Table 2. Performance in accuracy of the the supervised SVM, θ-mi-SVM and θ-MI-SVM for the train and test dataset

Majority bags	**SVM [acc.]**	**θ-mi-SVM [acc.]**		**θ-MI-SVM [acc.]**	
	inst.	*bags*	*inst.*	*bags*	*inst.*
train	95.85	-	91.06	-	91.62
test	86.85	82.61	76.94	83.33	82.89

7. CONCLUSION

In this work we have introduced two adaptations of well known solutions for the MIL problem, where it is formulated as a maximum margin problem. We move from the typical MIL assumption that bag labels are decided based on the presence of at least one positive instance in the bag, to an assumption where the bag labels are decided by the dominant instance label in the bag. This assumption is more reasonable to scenarios related to product and media content reviews, where the public's polarity towards the product is given by the dominant individual polarity. This new assumption invalidates the use of any MIL solution, and so we proposed two adaptations of the well known SVM for MIL solution, θ-mi-MIL and θ-MI-SVM, that fits this new assumption.

We tested θ-mi-MIL and θ-MI-SVM with the IMDb movie review corpus, and we showed that these have a comparable, although slightly poorer, performance compared to mi-SVM and MI-SVM solutions. Among other reasons, the decrease in performance can be a consequence of: 1) with our label assumptions there is more data ambiguity than with the MIL assumptions; 2) the proposed methods use only a portion of the dataset to estimate the model, because only some instances of each bag are picked out, according to the proposed methods; 3) the parameters θ_+ and θ_- were assumed to be 0.5, which might simply not be the optimal choice. Ideally this parameter would have been estimated.

In all the cases the models are shown to be robust when dealing with new data: there were only slight decreases of performance when dealing with unseen data.

Potential future work includes including the estimation of the parameters θ_+ and θ_- and further improving the formalization of the heuristics of the proposed methods.

8. REFERENCES

[1] Bing Liu, "Sentiment analysis and opinion mining," *Synthesis lectures on human language technologies*, vol. 5, no. 1, pp. 1–167, 2012.

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

[3] Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial intelligence*, vol. 89, no. 1, pp. 31–71, 1997.

[4] Bo Pang, Lillian Lee, and Shivakumar 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.

[5] Bo Pang and Lillian Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proceedings of the ACL*, 2004.

[6] Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng, "Reasoning with neural tensor networks for knowledge base completion," in *Advances in Neural Information Processing Systems*, 2013, pp. 926–934.

[7] Quoc V Le and Tomas Mikolov, "Distributed representations of sentences and documents.," in *ICML*, 2014, vol. 14, pp. 1188–1196.

[8] Oded Maron and Aparna Lakshmi Ratan, "Multiple-instance learn ing for natural scene classification.," in *ICML*, 1998, vol. 98, pp. 341–349.

[9] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann, "Support vector machines for multiple-instance learning," in *Advances in neural information processing systems*, 2002, pp. 561–568.

[10] Yoshua Bengio, Holger Schwenk, Jean-Sébastien Senécal, and Fréderic Morin, "Gauvain, jean-luc. neural probabilistic language models," *Innovations in Machine Learning*, pp. 137–186.

[11] Ronan Collobert and Jason Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 160–167.

[12] Andriy Mnih and Geoffrey E Hinton, "A scalable hierarchical distributed language model," in *Advances in neural information processing systems*, 2009, pp. 1081–1088.

[13] Joseph Turian, Lev Ratinov, and Yoshua Bengio, "Word representations: a simple and general method for semi-supervised learning," in *Proceedings of the 48th annual meeting of the association for computational linguistics*. Association for Computational Linguistics, 2010, pp. 384–394.

[14] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[15] Jeff Mitchell and Mirella Lapata, "Composition in distributional models of semantics," *Cognitive science*, vol. 34, no. 8, pp. 1388–1429, 2010.

[16] Fabio Massimo Zanzotto, Ioannis Korkontzelos, Francesca Fallucchi, and Suresh Manandhar, "Estimating linear models for compositional distributional semantics," in *Proceedings of the 23rd International Conference on Computational Linguistics*. Association for Computational Linguistics, 2010, pp. 1263–1271.

[17] Ainur Yessenalina and Claire Cardie, "Compositional matrix-space models for sentiment analysis," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011, pp. 172–182.

[18] Edward Grefenstette, Georgiana Dinu, Yao-Zhong Zhang, Mehrnoosh Sadrzadeh, and Marco Baroni, "Multi-step regression learning for compositional distributional semantics," *arXiv preprint arXiv:1301.6939*, 2013.

[19] T Mikolov and J Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, 2013.