Technical Report
EHU-KZAA-TR-4-2011

UNIVERSITY OF THE BASQUE COUNTRY
Department of Computer Science and Artificial Intelligence

Approaching Sentiment Analysis by Using Semi-supervised Learning of Multi-dimensional Classifiers

Jonathan Ortigosa-Hernández, Juan Diego Rodríguez, Leandro Alzate, Manuel Lucania, Iñaki Inza y Jose A. Lozano

February 2011

San Sebastian, Spain
http://www.ccia-kzaa.ehu.es/

Approaching Sentiment Analysis by Using Semi-supervised Learning of Multi-dimensional Classifiers

Jonathan Ortigosa-Hernández[a,*], Juan Diego Rodríguez[a], Leandro Alzate[b], Manuel Lucania[b], Iñaki Inza[a], Jose Antonio Lozano[a]

[a] *Intelligent Systems Group, Department of Computer Science and Artificial Intelligence, Computer Science Faculty, The University of the Basque Country, San Sebastián, Spain*
[b] *Socialware©, Bilbao, Spain*

Abstract

Sentiment Analysis is defined as the computational study of opinions, sentiments and emotions expressed in text. Within this broad field, most of the work has been focused on either Sentiment Polarity classification, where a text is classified as having positive or negative sentiment, or Subjectivity classification, in which a text is classified as being subjective or objective. However, in this paper, we consider instead a real-world problem in which the attitude of the author is characterised by three different (but related) target variables: Subjectivity, Sentiment Polarity, Will to Influence, unlike the two previously stated problems, where there is only a single variable to be predicted. For that reason, the (uni-dimensional) common approaches used in this area yield suboptimal solutions to this problem. In order to bridge this gap, we propose, for the first time, the use of the novel multi-dimensional classification paradigm in the Sentiment Analysis domain. This methodology is able to join the different target variables in the same classification task so as to take advantage of the potential statistical relations between them. In addition, and in order to take advantage of the huge amount of unlabelled information available nowadays in this context, we propose the extension of the multi-dimensional classification framework to the semi-supervised domain. Experimental results for this problem show that our semi-supervised multi-dimensional approach outperforms the most common Sentiment Analysis approaches, concluding that our approach is beneficial to improve the recognition rates for this problem, and in extension, could be considered to solve future Sentiment Analysis problems.

Keywords: Sentiment Analysis, Multi-dimensional Classification, Multi-dimensional Class Bayesian Network Classifiers, Semi-supervised Learning, EM Algorithm

1. Introduction

Sentiment Analysis (SA), which is also known as Opinion Mining, is a broad area defined as the computational study of opinions, sentiments and emotions expressed in text [29]. It mainly

*Corresponding author
Email addresses: jonathan.ortigosa@ehu.es (Jonathan Ortigosa-Hernández), juandiego.rodriguez@ehu.es (Juan Diego Rodríguez), leandro.alzate@asomo.net (Leandro Alzate), manuel.lucania@socialware.eu (Manuel Lucania), inaki.inza@ehu.es (Iñaki Inza), ja.lozano@ehu.es (Jose Antonio Lozano)

originated to meet the need for organisations to automatically find the opinions or sentiments of the general public about their products and services, as expressed on the Internet. A fundamental methodology in many current SA applications and problems is the well-known pattern recognition field called *classification* [5].

Most of the work within this field has focused on the *Sentiment Polarity classification*, i.e. determining if an opinionated text has positive or negative sentiment [34]. However, motivated by different real-world problems and applications, researchers have considered a wide range of closely related problems over a variety of different types of corpora [33]. As an example of these problems, we can find the following: *Subjectivity classification*, which consists of determining if a text is subjective or objective [37], *Authorship identification*, which deals with the problem of identifying the author of a given text [2] or *Affect Analysis*, which recognises emotions in a text [1].

In an analogous fashion, a real-world application within this field has recently been tackled in Socialware©[1], one of the most relevant companies in mobilised opinion analysis in Europe. The main goal of this application is to determine the attitude of the customers that write a post about a particular topic in a specific forum. The characterisation of these costumers is performed in this problem by measuring three different dimensions: the sentiment, the subjectivity and the potential influence of each post in the forum.

By just relying on the previous work done in the SA domain, we can approach this problem by dividing it into three different subproblems (one per each dimension to be classified) and tackle them separately, i.e. study them in isolation by learning different classifiers to predict the value of each target variable as if they were independent.

However, some of the individual approaches explored in the literature for each subproblem could be adapted to the others. Moreover, a large number of papers [31][50] proposed approaches for the problems of Sentiment Polarity classification and Subjectivity classification, and sometimes by even using the same corpus. This could indicate a certain degree of correlation between these subproblems, and consequently between their target variables. It is relatively easy to notice the relation that exists between the sentiment and subjectivity (a neutral review probably indicates objectivity). So, why not learn a single classifier to classify the three dimensions simultaneously so as to make use of the statistical similarities between them? Also, in extension to the SA domain, why not join some previously cited problems in the same classification task in order to find more accurate multi-dimensional classifiers that take advantage of their closeness?

In order to embody this perception, we propose the use of the recently proposed multi-dimensional Bayesian network classification framework [47] to deal with multiple class classification problems in the context of SA by solving a real-world application. This methodology performs a simultaneous classification by exploiting the relationships between the class variables to be predicted. Note that, under this framework, several target variables could be taken into account to enrich the SA problem and create market intelligence.

Furthermore, we are also concerned about the potential usage of the huge amount of unlabelled data available on the Internet. Most papers have already addressed the SA task by building classifiers that exclusively rely on labelled examples [29]. However, in practice, obtaining enough labelled examples for a classifier may be costly and time consuming, and this problem is accentuated when using multiple target variables. Thus, the scarcity of labelled data also motivates us to deal with unlabelled examples in a semi-supervised framework when working with the exposed multi-dimensional view.

[1] `http://www.asomo.net/indexE.jsp`

Motivated by the aforementioned comments, the following contributions are presented in this paper: (i) a novel and competitive methodology to solve the exposed real-world SA application, (ii) the use of multi-dimensional class Bayesian Network classifiers, in both supervised and semi-supervised frameworks, as a methodology to solve these multi-dimensional problems, and (iii) a innovate perspective to deal with the SA domain by dealing with several related problems in the same classification task.

The rest of the paper is organised as follows. Section 2 describes the real multi-dimensional problem extracted from the SA domain which is solved in this paper, reviews the work related to SA and its problems, and motivates the use of multi-dimensional approaches in this context. The multi-dimensional supervised classification paradigm is defined in Section 3. Section 4 describes the multi-dimensional class Bayesian network classifiers. A group of algorithms to learn different types of multi-dimensional Bayesian classifiers in a supervised framework is introduced in Section 5. Section 6 not only introduces the idea of semi-supervised learning into the multi-dimensional classification, but also extends the supervised algorithms presented in Section 5 to the semi-supervised framework. Section 7 shows the experimental results of applying the proposed multi-dimensional classification algorithms using different feature sets to the real problem stated in Section 3. Finally, Section 8 sums up the paper with some conclusions and future work recommendations.

2. Problem statement and state-of-the-art review

2.1. The Sentiment Analysis domain

The concept of SA, motivated by different real-world applications and business-intelligence requirements, has recently been interpreted more broadly to include many different types of analysis of text, such as the treatment of opinion, sentiment or subjectivity [33].

Within this broad field, the most known problem is referred to as *Sentiment Polarity classification*, in which the problem of classifying documents by their overall sentiment is considered, i.e. determining whether a review is positive or negative [34]. Several papers have expanded this original goal by, for instance, adding a neutral sentiment [50] or considering a multi-point scale [44] (e.g. one to five stars for a review) or using sentences or phrases instead of reviews as input of the sentiment classifier [50].

Work in Sentiment Polarity classification often assumes the incoming documents to be opinionated [33]. For many applications, though, we may need to decide whether a given document contains subjective information or not. This is referred to as *Subjectivity classification* and has gained considerable attention in the research community [37][38][48]. Due to its ability to distinguish subjective texts from the factual ones, it is also of great importance in SA. There are works in which a subjectivity classifier is used to filter the objective documents from a dataset before applying a sentiment classifier [50][31]. There are even works in which the need to predict both the Sentiment Polarity and the Subjectivity has been noticed [17]. As stated in [33], "the problem of distinguishing subjective versus objective instances has often proved to be more difficult than subsequent polarity classification, so improvements in subjectivity classification promise to positively impact sentiment classification".

Other closely related problems can be found in the SA domain: The set of problems called *Viewpoints and Perspectives* [33] which includes problems such as classifying political texts as liberal or conservative or placing texts along an ideological scale. *Authorship identification* deals with the problem of identifying the author of a given text [2]. *Affection Analysis* consists of extracting different types of emotions or affects from texts [1]. *Sarcasm Recognition* deals with the

SA hard-nut problem of recognising sarcastic sentences [46]. More problems and applications are discussed in [33].

2.2. The ASOMO problem: a multi-dimensional perspective of SA

In this paper, we deal with a recent real-world problem studied in Socialware©. This problem is extracted from its *ASOMO service* of mobilised opinion analysis and it has an underlying multi-dimensional nature that can be viewed as an augmentation of the classical problems of Sentiment Polarity classification and Subjectivity classification.

The main goal of this application is to determine the attitude of a customer when he writes a post about a particular topic in a specific forum through three different dimensions: Sentiment Polarity and Subjectivity (as widely used in the SA domain), and a third one called Will to Influence, which is frequently used in the framework of ASOMO. The latter is defined as the dimension that rates the desire of the opinion holder to cause a certain reaction in the potential readers of the text. While some people leave a post on a forum to tell of their experience, others seek to provoke a certain kind of reaction in the readers. In our application, this class variable has four possible values: none (declarative text), question (soft Will to Influence), complaint/recommendation (medium Will to Influence) and appellation (strong Will to Influence). We use two example cases in order to introduce the ASOMO problem:

- A customer who has bought an iPhone does not know how to set up 3G on the phone, so he writes on a forum: "How can I configure 3G on my iPhone?".
- Another customer is upset with the iPhone battery lifetime and writes in the same forum: "If you want long battery life, then don't buy an iPhone".

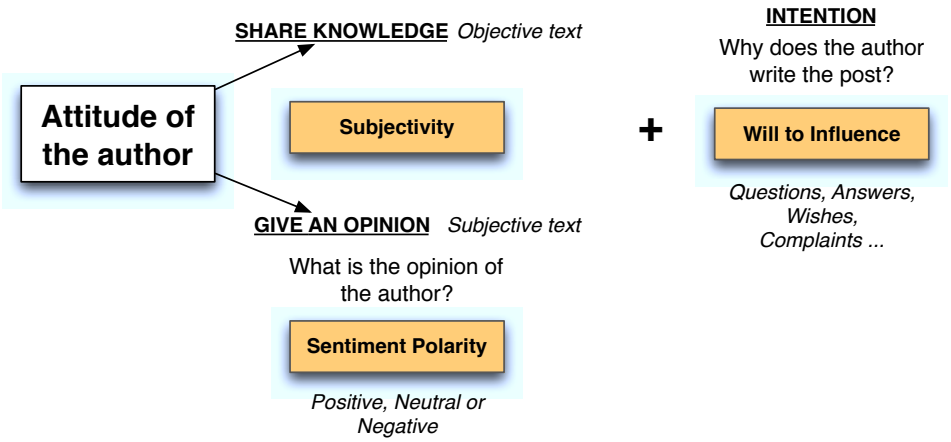
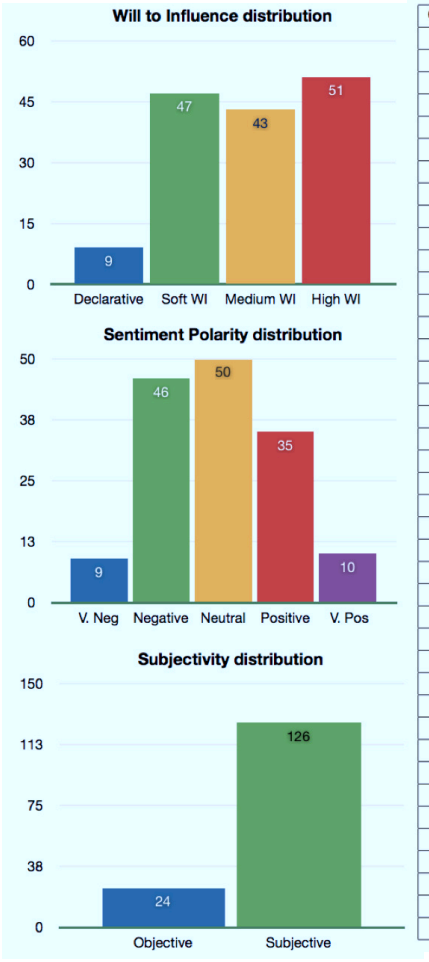


Figure 1: The vision of Socialware© company of the problem of determining the attitude of the writers in a forum.

The attitude of both customers is very different. The first one has a doubt and writes to obtain a solution to his problem {neutral sentiment, objective and soft Will to Influence} while the second writes so as not to recommend the iPhone {negative sentiment, subjective and strong Will to Influence}. Figure 1 shows one possible view of this problem and how we can translate the attitude of the author in three different class variables that are strongly correlated.

2.2.1. The ASOMO SA dataset

In order to deal with the previous problem, the ASOMO dataset was collected by Socialware©. This corpus was extracted from a single online discussion forum in which different customers of a specific company had left their comments, doubts and views about a single product. The forum consists of $2,542$ posts written in Spanish, with an average of 94 ± 74 words per post. 150 of these documents have been manually labelled by an expert in Socialware© according to the exposed three different dimensions and $2,392$ are left as unlabelled instances.



Conf.	Will to Influence	Sentiment Polarity	Subjectivity	Num
1	Declarative	Very Negative	No	0
2	Declarative	Very Negative	Yes	0
3	Declarative	Negative	No	0
4	Declarative	Negative	Yes	1
5	Declarative	Neutral	No	2
6	Declarative	Neutral	Yes	3
7	Declarative	Positive	No	0
8	Declarative	Positive	Yes	3
9	Declarative	Very Positive	No	0
10	Declarative	Very Positive	Yes	0
11	Soft WI	Very Negative	No	0
12	Soft WI	Very Negative	Yes	0
13	Soft WI	Negative	No	0
14	Soft WI	Negative	Yes	12
15	Soft WI	Neutral	No	12
16	Soft WI	Neutral	Yes	9
17	Soft WI	Positive	No	0
18	Soft WI	Positive	Yes	13
19	Soft WI	Very Positive	No	0
20	Soft WI	Very Positive	Yes	1
21	Medium WI	Very Negative	No	0
22	Medium WI	Very Negative	Yes	4
23	Medium WI	Negative	No	0
24	Medium WI	Negative	Yes	17
25	Medium WI	Neutral	No	3
26	Medium WI	Neutral	Yes	5
27	Medium WI	Positive	No	0
28	Medium WI	Positive	Yes	11
29	Medium WI	Very Positive	No	0
30	Medium WI	Very Positive	Yes	3
31	High WI	Very Negative	No	0
32	High WI	Very Negative	Yes	5
33	High WI	Negative	No	0
34	High WI	Negative	Yes	16
35	High WI	Neutral	No	7
36	High WI	Neutral	Yes	9
37	High WI	Positive	No	0
38	High WI	Positive	Yes	8
39	High WI	Very Positive	No	0
40	High WI	Very Positive	Yes	6
Total Number of Posts				150

Figure 2: Distribution of the labels of the three class variables over the labelled subset. The marginal distributions of Will to Influence, Sentiment Polarity and Subjectivity are represented as bar diagrams (left) and the joint distribution is represented in a table (right).

As previously mentioned, Will to Influence has four possible values: declarative text, soft, medium and strong will to influence. Sentiment Polarity, in this dataset, has 5 different labels as occurs in the 1-5 star ratings. In addition to the three classic values (positive, neutral and

negative), it has the values "very negative" and "very positive". Note that in using this approach, the label "neutral" in $\mathrm{Sentiment\ Polarity}$ is ambiguous, as happens in the SA literature [33]. So, it can be used as a label for the objective text (no opinion) or as a label for the sentiment that lies between positive and negative. As usual, $\mathrm{Subjectivity}$ has two values: objective and subjective. Figure 2 shows not only the label distribution in the dataset for each different class variable (the three bar diagrams on the left), but also the joint label distribution of these three class variables over the labelled subset of the ASOMO dataset (the table on the right). Note that there are configurations of the joint label distribution that are equal to zero, this is because there are configurations of the class variables which are not possible, e.g. $\{$strong Will to Influence, negative sentiment and objective$\}$.

As a result of the extensive work carried out by Socialware© on manually dealing with the ASOMO problem in the recent past, high levels of experience and understanding of determining the major factors that characterise the attitude of the customers have been gained. These factors are the following: (1) The implication of the author with the other customers in the forum, (2) the position of authority of the customer, and (3) the subjective language used in the text.

These broad factors have been helpful in detecting a list of 14 morphological features which characterise each analysed document. In order to engineer this list of features, each document is preprocessed using an open source morphological analyser [3][7]. Firstly, spelling in the entire corpus is checked. Then, the analyser provides information related to the part-of-the-speech (PoS) tagging [34]. Once the preprocessing task is performed, determining the values of the features is carried out by just looking for specific patterns in the corpus. In the following paragraphs, a detailed introduction of each factor is given, as well as a description of the features used in each factor.

	Feature	Description	Example	Translation
1	*First Persons*	Number of verbs in the fist person.	Contraté ...	I hired ...
2	*Second Persons*	Number of verbs in the second person.	Tienes ...	You have ...
3	*Third Persons*	Number of verbs in the third person.	Sabe ...	He knows ...
4	*Relational Forms*	Number of phatic expressions, i.e. expressions whose only function is to perform a social task.	(1) Hola. (2) Gracias de antemano.	(1) Hello. (2) Thanks in advance.
5	*Agreement Expressions*	Number of expressions that show agreement or disagreement.	(1) Estoy de acuerdo contigo. (2) No tienes razón.	(1) I agree with you. (2) You're wrong.
6	*Request*	Number of sentences that express a certain degree of request.	(1) Me gustaría saber ... (2) Alguien podría ...	(1) I'd like to know ... (2) I would appreciate it if someone could ...

Table 1: Subset of features related to the implication of the author with other customers.

The implication of the author. This factor covers the features that are related with the interaction between the author and the other customers in the forum. It consists of six different features that are described in Table 1. For each feature, we show its description and an example (with its translation into English) of the type of pattern that matches with the feature.

The position of authority of the opinion holder is mainly characterised by the purpose of the written post and it is related to the potential influence on the readers of the forum. The author could express advice, disapproval with a specific product, predictions, etc. Table 2 shows the six features that are part of this major factor.

	Feature	Description	Example	Translation
7	*Imperatives*	Number of imperative verbs in the second person.	No compres.	Do not buy
8	*Exhorts and Advice*	Number of exhort verbs, e.g. recommend, advise, prevent, etc.	Te recomiendo ...	I recommend that you ...
9	*Sufficiency Expressions*	Number of expressions used to corroborate other sentences of the text.	(1) Por supuesto. (2) Naturalmente, ...	(1) Of course. (2) Naturally, ...
10	*Prediction Verbs*	Number of verbs in the future.	(1) Voy a probar. (2) Llamaré.	(1) I'm going to try. (2) I'll call.
11	*Authority*	Number of expressions that denote high degree of authority, usually written in the subjunctive mode	Si fuera tú, ...	If I were you, ...
12	*Questions*	Number of question in the post, both direct and indirect.	(1) ¿Qué tal es? (2) Dime qué te parece.	(1) How is it? (2) Tell me what you think of it.

Table 2: Subset of features related to the position of authority of the customer.

Subjective language deals with the opinion of the author. In order to determine this factor, we consider only the adjective detected with the PoS recogniser, as commonly carried out in the state-of-the-art literature [23]. Then, the adjectives are classified in polarity terms by means of a hand-annotated sentiment-lexicon. As a result of this task, we obtain two features: *Positive Adjectives* and *Negative Adjectives*, which are the number of positive and negative adjectives, respectively, in the text.

The 14 features (the ASOMO features) are normalised to be in the range $[0, 1]$ by dividing them by the maximal observed value.

2.3. The need for multi-dimensional classification techniques

In order to solve the typical problems of the SA domain (those exposed in Section 2.1), there are two main types of techniques that can be distinguished: *Symbolic* and *Machine Learning* [16]. The symbolic approach uses manually crafted rules and lexicons, whereas the machine learning approach uses supervised or semi-supervised learning to construct a model from a training corpus. Due to the fact that the main proposal of this paper is to solve a real SA problem by means of a novel machine learning technique, this review focuses on the latter. The machine learning approaches have gained interest because of (1) their capability to model many features and, in doing so, capturing context, (2) their easier adaptability to changing input, and (3) the possibility to measure the degree of uncertainty by which a classification is made [16]. Supervised methods that train from examples which have been manually classified by humans are the most popular.

Most of the work that has been carried out in tuning up these machine learning techniques (as also happens in text processing tasks) has been dedicated to addressing the problem of converting

a piece of text into a feature vector (i.e. model features able to capture the context of the text) in order to improve the recognition rates. The most common approaches use the single lower cased words (unigrams) as features, which in several cases reports pretty good results as in [34]. However, other common approaches can be found, such as n-grams [31] or PoS information. A deeper study of such work is beyond the scope of this paper. The reader who is interested in feature engineering can consult [18], where there is an extensive body of work that addresses feature selection for machine learning approaches in general.

On the other hand, less research has been done on the induction of the classifiers. Most of the existing works learn either a naive Bayes [34][50] or a support vector machine (SVM) [2][34], i.e. uni-dimensional classifiers able to predict a single target variable. For that reason, the classification models used in the SA literature seem inappropriate to model the three-dimensional problem exposed in this paper. However, there are several possibilities to adapt uni-dimensional classifiers to multi-dimensional classification problems, and the ASOMO problem is no exception. Unfortunately, none of these approaches captures exactly the underlying characteristics of the problem [40]:

- One approach is to develop multiple classifiers, one for each class variable. However, this approach does not capture the real characteristics of the problem, because it does not model the correlations between the different class variables and so, it does not take advantage of the information that they may provide. It treats the class variables as if they were independent. In the case of the previously exposed problem, it would be splitting it into three different uni-dimensional problems, one per each class variable.

- Another approach consists of constructing a single artificial class variable that models all possible combinations of classes. This class variable models the Cartesian product of all the class variables. The problem of this approach arises because this compound class variable can easily end up with an excessively high cardinality. This leads to computational problems because of the high number of parameters the model has to estimate. Furthermore, the model does not reflect the real structure of the classification problem either. By means of this approach, the ASOMO problem would be redefined as a uni-dimensional problem with a 40-label class variable.

The previous approaches are clearly insufficient and suboptimal for the resolution of problems where class variables have high cardinalities or large degrees of correlation among them. The first approach does not reflect the multi-dimensional nature of the problem because it does not take into account any correlation among the class variables. The second approach, however, does not consider the possible conditional independences between the classes and assumes models that are too complex. As can be seen in the experiments section, these deficiencies in capturing the real relationship between the class variables may cause a low performance, so new techniques are required to bridge the gap between the solutions offered by the learning algorithms used in the SA literature and the multi-dimensional underlying nature of the ASOMO problem.

Within this framework, multi-dimensional classification techniques appear in order to yield more adequate models which are able to use the correlations and conditional independencies between class variables in order to help in the classification task in both supervised and semi-supervised learning frameworks.

3. Multi-dimensional classification

In this section we present, in detail, the nature of the multi-dimensional supervised classification paradigm and how to define and evaluate a multi-dimensional classifier.

3.1. Multi-dimensional supervised classification problems

A typical (uni-dimensional) supervised classification problem consists of building a classifier from a labelled training dataset (see Table 3) in order to predict the value of a class variable C given a set of features $\mathbf{X} = (X_1, ..., X_n)$ of an unseen unlabelled instance $\mathbf{x} = (x_1, ..., x_n)$.

If we suppose that $(\mathbf{X}, C)$ is a random vector with a joint feature-class probability distribution $p(\mathbf{x}, c)$ then, a classifier ψ is a function that maps a vector of features $\mathbf{X}$ into a single class variable C:

$$\begin{array}{rcl} \psi : \{0, ..., r_1 - 1\} \times ... \times \{0, ..., r_n - 1\} & \mapsto & \{0, ..., t - 1\} \\ \mathbf{x} & \mapsto & c \end{array}$$

where r_i and t are the number of possible values of each feature $X_i, (i = 1, ..., n)$ and the class variable respectively.

X_1	X_2	...	X_n	C
$x_1^{(1)}$	$x_2^{(1)}$	...	$x_n^{(1)}$	$c^{(1)}$
$x_1^{(2)}$	$x_2^{(2)}$	...	$x_n^{(2)}$	$c^{(2)}$
...	...	...	...	...
$x_1^{(N)}$	$x_2^{(N)}$	...	$x_n^{(N)}$	$c^{(N)}$

Table 3: A possible representation of a (uni-dimensional) labelled training dataset.

X_1	X_2	...	X_n	C_1	C_2	...	C_m
$x_1^{(1)}$	$x_2^{(1)}$	...	$x_n^{(1)}$	$c_1^{(1)}$	$c_2^{(1)}$	...	$c_m^{(1)}$
$x_1^{(2)}$	$x_2^{(2)}$	...	$x_n^{(2)}$	$c_1^{(2)}$	$c_2^{(2)}$	...	$c_m^{(2)}$
...	...	...	...	...	...	...	...
$x_1^{(N)}$	$x_2^{(N)}$	...	$x_n^{(N)}$	$c_1^{(N)}$	$c_2^{(N)}$	...	$c_m^{(N)}$

Table 4: Representation of a multi-dimensional labelled training dataset.

A generalisation of this problem to the simultaneous prediction of several class variables has recently been proposed in the research community [4][14][36][39][40][47]. This generalisation is known as multi-dimensional supervised classification. Its purpose is to simultaneously predict the value of each class variable in the class variable vector $\mathbf{c} = (c_1, ..., c_m)$ given the feature vector $\mathbf{x} = (x_1, ..., x_n)$ of an unseen unlabelled instance. The training dataset, in this multi-dimensional framework, is expressed as shown in Table 4.

Thus, the classifier ψ becomes a function that maps a vector of features $\mathbf{X}$ into a vector of class variables $\mathbf{C}$:

$$\begin{array}{rcl} \psi : \{0, ..., r_1 - 1\} \times ... \times \{0, ..., r_n - 1\} & \mapsto & \{0, ..., t_1 - 1\} \times ... \times \{0, ..., t_m - 1\} \\ \mathbf{x} & \mapsto & \mathbf{c} \end{array}$$

where r_i and t_j are the cardinalities of each feature X_i (for $i = 1, ..., n$) and each class variable C_j (for $j = 1, ..., m$) respectively. Note that we consider all variables, both predictive features and class variables, as discrete random variables.

A classifier is learnt from a training set (see Table 4) with a classifier induction algorithm $A(\cdot)$. Given the induction algorithm $A(\cdot)$, which is assumed to be a deterministic function of the training set, the multi-dimensional classifier obtained from a training set D is denoted as $\psi = A(D)$.

3.2. Multi-dimensional classification rule

In probabilistic classification, the induction algorithm learns a probability distribution $p(\mathbf{x}, \mathbf{c})$ or $p(\mathbf{c}|\mathbf{x})$ from the training data and classifies a new unlabelled instance based on it. For that purpose, a classification rule must be defined.

In uni-dimensional supervised classification, the most common classification rule returns the most likely class value given the features:

$$\hat{c} = argmax_c \{p(c|x_1, \cdots, x_n)\}$$

The multi-dimensional nature of the problem allows us to develop several classification rules that would make no sense in single-class classification because they take into account multiple class variables. Nevertheless, the previous one-dimensional classification rule can be easily generalised to the prediction of more than one class variable. In this case, the multi-dimensional classifier returns the most probable combination of class variables given the features. This rule is known as *joint classification rule* [40]:

$$(\hat{c}_1, \cdots, \hat{c}_m) = argmax_{c_1,\cdots,c_m} \{p(c_1, \cdots, c_m|x_1, \cdots, x_n)\}$$

Although several other classification rules are proposed in [40], it is shown that the joint classification rule obtains better results.

3.3. Multi-dimensional classification evaluation

Once a classifier is constructed, its associated error needs to be measured. The prediction error of a single-class classifier ψ is the probability of the incorrect classification of an unlabelled instance $\mathbf{x}$ and is denoted as $\epsilon(\psi)$:

$$\epsilon(\psi) = p(\psi(\mathbf{X}) \neq C) = E_{\mathbf{X}}[\delta(c, \psi(\mathbf{x}))]$$

where $\delta(x, y)$ is a loss function whose results are 1 if $x \neq y$ and 0 if $x = y$.

However, in multi-dimensional classification, the correctness of a classifier can be measured in two different ways:

- Joint evaluation: This consists of evaluating the estimated values of all class variables simultaneously, that is, it only counts a success if all the classes are correctly predicted, and otherwise it counts an error:

  $$\epsilon(\psi) = p(\psi(\mathbf{X}) \neq \mathbf{C}) = E_{\mathbf{X}}[\delta(\mathbf{c}, \psi(\mathbf{x}))]$$

  This rule is the generalisation of the previous single-class evaluation measure to multi-dimensional classification.

- Single evaluation: After a multi-dimensional learning process, this consists of separately checking if each class is correctly classified. For example, if we classify an instance $\mathbf{x}$ as

$(\hat{c}_1 = 0, \hat{c}_2 = 1)$ and the real value is $(c_1 = 0, c_2 = 0)$, we count $\hat{c}_1$ as a success and $\hat{c}_2$ as an error. This approach provides one performance measure for each class C_j (for $j = 1, \cdots, m$). The output of this evaluation is a vector $\boldsymbol{\epsilon}$ of size m with the performance function of the multi-dimensional classifier for each of the class variables:

$$\epsilon_j(\psi) = p(\psi_j(\mathbf{X}) \neq C_j) = E_{\mathbf{X}}[\delta(c_j, \psi_j(\mathbf{x}))]$$

where $\psi_j(\mathbf{x})$ is the estimation of the multi-dimensional classifier for the j-th class variable.

Ideally, we would like to exactly calculate the error of a classifier, but in most real world problems the feature-label probability distribution $p(\mathbf{x}, \mathbf{c})$ is unknown. So, the prediction error of a classifier ψ is also unknown; it can not be computed exactly, and thus, must be estimated from data.

Several approaches to estimate the prediction error can be used. In this work, we use one of the most popular error estimation techniques: k-fold cross-validation (k-cv) [45] in its repeated version. In k-cv the dataset is divided into k folds, a classifier is learnt using $k - 1$ folds and an error value is calculated by testing the learnt classifier in the remaining fold. Finally, the k-cv estimation of the error is the average value of the errors made in each fold. The repeated r times k-cv consists of estimating the error as the average of r k-cv estimations with different random partitions into folds. This method considerably reduces the variance of the error estimation [41].

In multi-dimensional classification we could be interested in either learning the most accurate classifier for all class variables simultaneously (measured with a joint evaluation) or in finding the most accurate classifier for each single class variable (measured with single evaluations). In this paper, we are mainly interested in using the joint evaluation for evaluation. However, in our application to SA we also measure the performance of the algorithms with a single evaluation per each class variable in order to compare both types of evaluation and perform a deeper analysis of the results.

4. Multi-dimensional Bayesian network classifiers

In this section, multi-dimensional class Bayesian network classifiers [4][47], which are able to deal with multiple class variables to be predicted, are presented as a recent generalisation of the classical Bayesian network classifiers [28].

4.1. Bayesian network classifiers

Bayesian networks are powerful tools for knowledge representation and inference under uncertainty conditions [35]. These formalisms have been extensively used as classifiers [27] and have become a classical and well-known classification paradigm.

A Bayesian network is a pair $B = (S, \boldsymbol{\Theta})$ where S is a directed acyclic graph (DAG) whose vertices correspond to random variables and whose arcs represent conditional (in)dependence relations among variables, and where $\boldsymbol{\Theta}$ is a set of parameters.

A Bayesian classifier is usually represented as a Bayesian network with a particular structure. The class variable is on the top of the graph and it is the parent of all predictive variables.

In spite of the popularity of Bayesian network classifiers, few works have taken into account their generalisation to multiple class variables[4][14][39][40][47]. In multi-dimensional classification, we consider Bayesian networks over a finite set $\mathbf{V} = \{C_1, \cdots, C_m, X_1, \cdots, X_n\}$ where each class variable C_j and each feature X_i takes a finite set of values. $\boldsymbol{\Theta}$ is formed by parameters

π_{ijk} and θ_{ijk}, where $\pi_{ijk} = p(C_i = c_k|\mathbf{Pa}(C_i) = \mathbf{Pa}(c_i)_j)$ for each value c_k that can take each class variable C_i and for each value assignment $\mathbf{Pa}(c_i)_j$ to the set of the parents of C_i. Analogously, $\theta_{ijk} = p(X_i = x_k|\mathbf{Pa}(X_i) = \mathbf{Pa}(x_i)_j)$ for each value x_k that can take each feature X_i and for each value assignment $\mathbf{Pa}(x_i)_j$ to the set of the parents of X_i.

Thus, the network B defines a joint probability distribution $p(c_1, \cdots, c_m, x_1, \cdots, x_n)$ given by:

$$p(c_1, \cdots, c_m, x_1, \cdots, x_n) = \prod_{i=1}^{m} \pi_{ijk} \prod_{i=1}^{n} \theta_{ijk}$$

4.2. Structure of multi-dimensional class Bayesian network classifiers

A multi-dimensional class Bayesian network classifier is a generalisation of the classical one-class variable Bayesian classifiers for domains with multiple class variables [47]. It models the relationships between the variables by means of directed acyclic graphs (DAG) over the class variables and over the feature variables separately, and then connects the two sets of variables by means of a bi-partite directed graph. So, the DAG structure $S = (\mathbf{V}, \mathbf{A})$ has the set $\mathbf{V}$ of random variables partitioned into the sets $\mathbf{V}_C = \{C_1, \ldots, C_m\}$, $m > 1$, of class variables and the set $\mathbf{V}_F = \{X_1, \ldots, X_n\}$ ($n \geq 1$) of features. Moreover, the set of arcs $\mathbf{A}$ can be partitioned into three sets: $\mathbf{A}_{CF}$, $\mathbf{A}_C$ and $\mathbf{A}_F$ with the following properties:

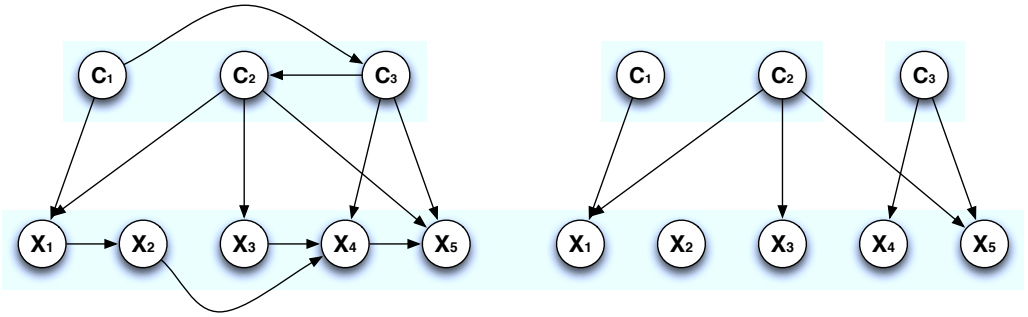
- $\mathbf{A}_{CF} \subseteq \mathbf{V}_C \times \mathbf{V}_F$ is composed of the arcs between the class variables and the feature variables, so we can define the feature selection subgraph of S as $S_{CF} = (\mathbf{V}, \mathbf{A}_{CF})$. This subgraph represents the selection of features that seems relevant for classification given the class variables.
- $\mathbf{A}_C \subseteq \mathbf{V}_C \times \mathbf{V}_C$ is composed of the arcs between the class variables, so we can define the class subgraph of S induced by $\mathbf{V}_C$ as $S_C = (\mathbf{V}_C, \mathbf{A}_C)$.
- $\mathbf{A}_F \subseteq \mathbf{V}_F \times \mathbf{V}_F$ is composed of the arcs between the feature variables, so we can define the feature subgraph of S induced by $\mathbf{V}_F$ as $S_F = (\mathbf{V}_F, \mathbf{A}_F)$.

Figure 3 shows a multi-dimensional class Bayesian network classifier with 3 class variables and 5 features, and its partition into the three subgraphs.

Depending on the structure of the three subgraphs, the following sub-families[2] of multi-dimensional class network classifiers are proposed in the state-of-the-art literature:

- *Multi-dimensional naive Bayes classifier (MDnB)*: the class subgraph and the feature subgraph are empty and the feature selection subgraph is complete.
- *Multi-dimensional tree-augmented Bayesian network classifier (MDTAN)*: both the class subgraph and the feature subgraph are directed trees. It could be viewed as the multi-dimensional version of the (uni-dimensional) tree-augmented Bayesian network classifier (TAN) proposed in [20].
- *Multi-dimensional J/K dependences Bayesian classifier (MD J/K)*: This structure is the multi-dimensional generalisation of the well-known K-DB [43] classifier. It allows each class variable C_i to have a maximum of J dependences with other class variables C_j, and each predictive variable X_i to have, apart from the class variables, a maximum of K dependences with other predictive variables.

[2] In [47], [14] and [39], instead of *multi-dimensional*, the term *fully* is used in order to name the classifiers.



(a) Complete graph

(b) Feature selection subgraph

(c) Class subgraph

(d) Feature subgraph

Figure 3: A multi-dimensional Bayesian classifier and its division.

In the following section, several algorithms are provided in order to learn from a given dataset the previous sub-families of multi-dimensional Bayesian network classifiers.

5. Learning multi-dimensional Bayesian network classifiers

As in the classic Bayesian network learning task, learning a multi-dimensional class Bayesian network classifier from a training dataset consists of estimating its structure and its parameters. These two subtasks are called structure learning and parameter learning respectively. Due to the fact that each previously introduced sub-family has different restrictions in its structure, a different learning algorithm is needed for each one.

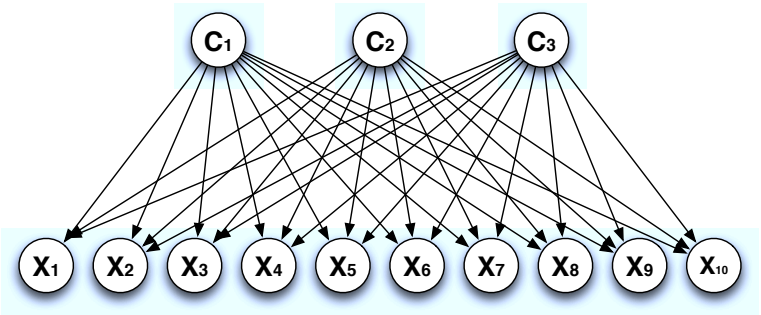


Figure 4: An example of a multi-dimensional naive Bayes structure.

In this section, we provide algorithms for learning MDnB [47], MDTAN [47] and MD J/K classifiers from a given dataset. The MDnB and MD J/K learning algorithms use a filter approach, i.e. the learning task precedes the classification evaluation. However, as it is proposed in its original work [47], the MDTAN learning algorithm is formulated as a wrapper approach [21], i.e. it tries to find more accurate classifiers by taking advantage of the classification evaluation.

As Figure 4 shows, in the MDnB classifier, each class variable is parent of all the features, and each feature has only all the class variables as parents. Conventionally, the class subgraph and the feature subgraph are empty and the feature selection subgraph is complete. This classifier assumes conditional independence between each pair of features given the entire set of class variables. Due to the fact that it has no structure learning (the structure is fixed for a determined number of class variables and features), learning a MDnB classifier consists of just estimating the parameters Θ of the actual model by using a training dataset D. This is achieved by calculating the maximum likelihood estimator (MLE) [13].

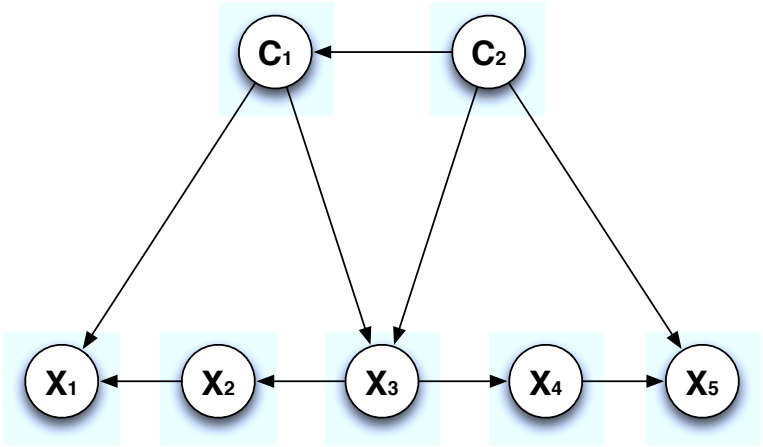


Figure 5: An example of a multi-dimensional tree-augmented network structure.

Instead, learning a MDTAN classifier consists of learning both structure and parameters. A wrapper structure learning algorithm is proposed in [47]. Its aim is to produce the MDTAN structure that maximises the accuracy from a given dataset. This algorithm has a main part called *Feature subset selection algorithm*, which follows a wrapper approach [21] by performing a local search over the A_{CF} structure. In order to obtain a MDTAN structure in each iteration, it generates a set of different A_{CF} structures from a current A_{CF} and learns its class subgraph and feature subgraph by using the following sub-algorithms:

1. A_C *structure learning algorithm* is the algorithm that learns the structure between the class variables by building a maximum weighted spanning [25] tree using mutual information.
2. A_F *structure learning algorithm* learns the A_F subgraph by using conditional mutual information, by means of the Chow and Liu algorithm [9].

After that, the accuracies of all the learnt models are computed. The iterative process continues by setting the A_{CF} of the best classifier, in terms of estimated accuracy, as current A_{CF}. This algorithm belongs to the hill-climbing family of optimisation algorithms, i.e. when no improvement is achieved by generating a new set of structures, the algorithm stops.

5.1. Multi-dimensional J/K dependences Bayesian classifier

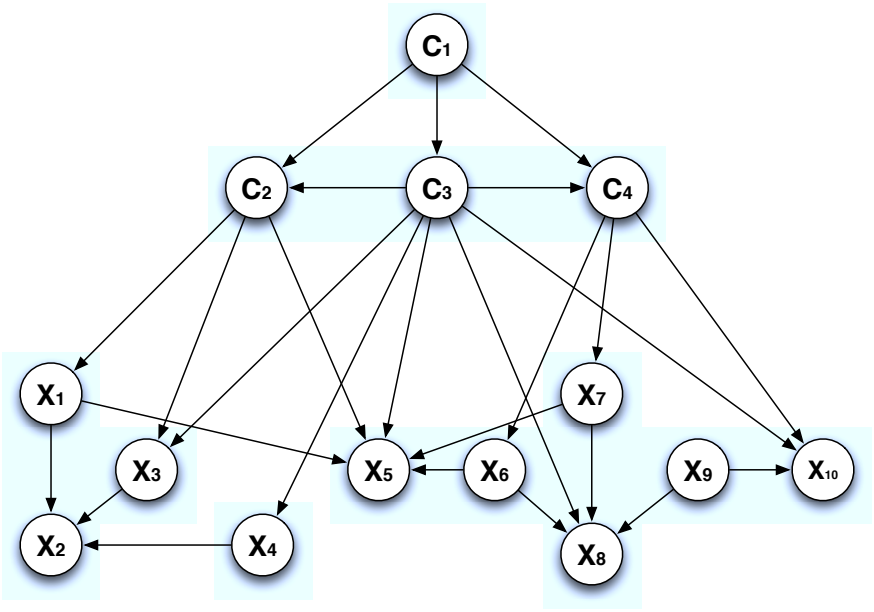


Figure 6: An example of a multi-dimensional 2/3-dependence Bayesian network structure.

The MD J/K (see Figure 6), which is introduced in [40], is the generalisation of the K dependence structure [43] to the multi-dimensional framework. It is able to move through the spectrum of allowable dependence in the multi-dimensional framework, from the MDnB to the full multi-dimensional Bayesian classifier. Note that from the setting $J = K = 0$, we can learn a MDnB, setting $J = K = 1$ a MDTAN structure is learned and so on. The full multi-dimensional Bayesian classifier, which is the classifier that has the three complete subgraphs, can be learnt by setting $J = (m - 1)$ and $K = (n - 1)$, where m and n are the number of class variables and predictive features respectively.

Although the MD J/K structure has been proposed in the state-of-the-art literature, to the best of our knowledge, a specific MD J/K learning algorithm for the multi-dimensional framework has not been defined by the research community. In this section, we propose a filter algorithm in a supervised learning framework capable of learning this type of structure (see Algorithm 1).

In this algorithm, we do not directly use the mutual information as measured in the previous MDTAN learning algorithm [47]. This is due to the fact that the mutual information is not normalised when the cardinalities of the variables are different, so we use an independence test to determine if a dependence between two variables is strong enough to be part of the model: It is known [26] that $2N\hat{I}(X_i, X_j)$ asymptotically follows a χ^2 distribution with $(r_i - 1)(r_j - 1)$ degrees of freedom, where N is the number of cases,, if X_i and X_j are independent, i.e. $Lim_{N\to\infty} 2N\hat{I}(X_i, X_j) \rightsquigarrow \chi^2_{(r_i-1)(r_j-1)}$.

Based on this result, a statistical hypothesis test can be carried out in a multi-dimensional Bayesian network classifier to check the robust dependences in A_C. The null hypothesis H_0 is

that the random variables C_i and C_j are independent. If the quantity $2N\hat{I}(C_i, C_j)$ surpasses a threshold s_α for a given test size

$$\alpha = \int_{s_\alpha}^{\infty} \chi^2_{(t_i-1)(t_j-1)} ds,$$

where t_i is the cardinality of C_i and t_j the cardinality of C_j, the null hypothesis is rejected and a dependence between C_i and C_j is considered. Therefore the arc between these class variables is included in the model. The dependences in A_{CF} are calculated using the same procedure, the null hypothesis H_0 is that "The random variables C_i and X_j are independent". So, if $2N\hat{I}(C_i, X_j)$ surpasses the threshold s_α, then the null hypothesis is rejected and an arc is included in the model. This test was also used on single-class Bayesian network classifiers to check the dependences among the class variables and the features [6].

Using this approach, the structures A_C and A_{CF} are learnt in steps 1 and 2 of Algorithm 1, respectively.

Algorithm 1 A MD J/K structure learning algorithm using a filter approach

1: **Learn the A_C structure**

1. Calculate the p-value using the independence test for each pair of class variables, and rank them.
2. Remove the p-value higher than the threshold $(1 - s_\alpha) = 0.10$.
3. Use the ranking to add arcs between the class variables fulfilling the conditions of no cycles between the class variables and no more than J-parents per class.

2: **Learn the A_{CF} structure**

1. Calculate the p-value using the independence test for each pair C_i and X_j and rank them.
2. Remove the p-value higher than the threshold $(1 - s_\alpha) = 0.10$.
3. Use the ranking to add arcs from the class variables to the features.

3: **Learn the A_F structure**

1. Calculate the p-value using the conditional independence test for each pair X_i and X_j given $\mathbf{Pa_c}(X_j)$ and rank them.
2. Remove the p-value higher than the threshold $(1 - s_\alpha) = 0.10$.
3. Use the ranking to add arcs between the class variables fulfilling the conditions of no cycles between the features and no more than K-parents per feature.

In order to calculate the structure A_F, we need to use the conditional mutual information between a feature X_i and a feature X_j given its class parents $\mathbf{Pa_c}(X_j)$ to determine if the relation between both predictive features should be included in the model. For that purpose, we use the generalisation of the previous result to the case of conditional mutual information as defined in [26]:

$$Lim_{N\to\infty} 2N\hat{I}(X_i, X_j|\mathbf{Pa_c}(X_j)) \rightsquigarrow \chi^2_{(r_i-1)(r_j-1)(|\mathbf{Pa_c}(X_j)|)}$$

where r_i is the cardinality of X_i, r_j the cardinality of X_j and $|\mathbf{Pa_c}(X_j)|$ the cardinality of the class parents of X_j.

Analogously to the hypothesis test previously described, based on these results we can perform the following conditional independence test: The null hypothesis assumes that the random variables X_i and X_j are conditionally independent given $\mathbf{Pa_c}(X_j)$. So, if the quantity $2N\hat{I}(C_i, C_j|\mathbf{Pa_c}(X_j))$ surpasses a threshold s_α for a given test size

$$\alpha = \int_{s_\alpha}^{\infty} \chi^2_{(t_i-1)(t_j-1)(|\mathbf{Pa_c}(X_j)|)} ds,$$

the null hypothesis is rejected and the random variables X_i and X_j are considered dependent given $\mathbf{Pa_c}(X_j)$. Therefore, the arc is included in the model. The structure A_{CF} is learnt using this hypothesis test in step 3 of Algorithm 1.

6. Semi-supervised multi-dimensional classification

In this section, we proceed with the extension of the previous multi-dimensional learning algorithms to the semi-supervised learning framework. When large amounts of labelled data are available, one can apply familiar and powerful machine learning techniques such as the previous multi-dimensional Bayesian network algorithms in order to learn accurate classifiers. However, when there is a scarcity of such labelled data and a huge amount of unlabelled data, as happens in the SA domain, one can wonder if it is possible to learn competitive classifiers from unlabelled data.

	X_1	X_2	...	X_n	C_1	C_2	...	C_m
D_L	$x_1^{(1)}$	$x_2^{(1)}$	...	$x_n^{(1)}$	$c_1^{(1)}$	$c_2^{(1)}$	...	$c_m^{(1)}$
	$x_1^{(2)}$	$x_2^{(2)}$	...	$x_n^{(2)}$	$c_1^{(2)}$	$c_2^{(2)}$	...	$c_m^{(2)}$
	...	...	...	...	...	...	...	...
	$x_1^{(L)}$	$x_2^{(L)}$	...	$x_n^{(L)}$	$c_1^{(L)}$	$c_2^{(L)}$	...	$c_m^{(L)}$
D_U	$x_1^{(L+1)}$	$x_2^{(L+1)}$	...	$x_n^{(L+1)}$	?	?	...	?
	$x_1^{(L+2)}$	$x_2^{(L+2)}$	...	$x_n^{(L+2)}$	?	?	...	?
	...	...	...	...	...	...	...	...
	$x_1^{(N)}$	$x_2^{(N)}$	...	$x_n^{(N)}$	?	?	...	?

Table 5: A formal representation of a multi-dimensional semi-supervised training dataset.

In this context, where the training dataset consists of labelled and unlabelled data, the semi-supervised learning approach [8][51][52] appears as a promising alternative. It is motivated from the fact that in many real world problems, obtaining unlabelled data is relatively easy, e.g. collecting posts from different blogs, while labelling is expensive and/or labor intensive, due to the fact that the tasks of labelling the training dataset is usually carried out by human beings. Thus, it is highly desirable to have learning algorithms that are able to incorporate a large number of unlabelled data with a small number of labelled data when learning classifiers.

In the semi-supervised learning framework, the training dataset D, as shown in Table 5, is divided into two parts: the subset of instances D_L for which labels are provided, and the subset D_U, where the labels are not known. Therefore, we have a dataset of N instances, where there

are L labelled examples and $(N - L)$ unlabelled examples. Normally, $(N - L) >> L$, i.e. the unlabelled subset tends to have a very large amount of instances whilst the labelled subset tends to have a small size.

Therefore, the aim of a semi-supervised learning algorithm is to build more accurate classifiers using both labelled and unlabelled data, rather than using exclusively labelled examples as happens in supervised learning.

6.1. Learning multi-dimensional Bayesian network classifiers in the semi-supervised framework

In this section, we propose the extension of multi-dimensional Bayesian network classifiers to the semi-supervised learning framework by using the EM algorithm [15]. Although this method was proposed in [15] and was deeply analysed in [30], it had been used much earlier, e.g. [22], and it is still widely used in many recent semi-supervised learning algorithms, e.g. [11][12][32].

The aim of the EM algorithm as typically used in semi-supervised learning is to find the parameters of the model that maximise the likelihood of the data, using both labelled and unlabelled instances. The iterative process works as follows: in the K-th iteration the algorithm alternates between completing the unlabelled instances by using the parameters $\mathbf{\Theta}^{(K)}$ (E-step) and updating the parameters of the model $\mathbf{\Theta}^{(K+1)}$ using MLE with the whole dataset (M-step), i.e. the labelled data and the unlabelled instances that have been previously classified in the E-Step. Note that the structure remains fixed in the whole iterative process.

Although good results have been achieved with the EM algorithm in uni-dimensional classification [11][32], we are concerned about the restriction of only maximising the parameters of a fixed structure in our extension of the EM algorithm to the multi-dimensional domain, where there are several class variables to be predicted. As stated in [10], if the correct structure of the real distribution of the data is obtained, unlabelled data improve the classifier, otherwise, unlabelled data can actually degrade performance. For this reason, it seems more appropriate to perform a structural search in order to find the real model. Thus, we perform several changes to the EM algorithm in order to avoid fixing the structure of the model during the iterative process. The proposal is shown in Algorithm 2.

Algorithm 2 Our version of the EM Algorithm

Input: A training dataset with both labelled and unlabelled data (Table 5) and an initial model $\psi^{(K=0)}$ with a fixed structure and with an initial set of parameters $\mathbf{\Theta}^{(K=0)}$.

1: **while** the model $\psi^{(K)}$ does not converge **do**
2: **E-STEP** Use the current model $\psi^{(K)}$ to estimate the probability of each configuration of class variables for each unlabelled instance.
3: **M-STEP** Learn a new model $\psi^{(K+1)}$ with structure and parameters, given the estimated probabilities in the E-STEP.
4: **end while**

Output: A classifier ψ, that takes an unlabelled instance and predicts the class variables.

In this version of the EM algorithm, we want to find the model, both structure and parameters, that maximises the likelihood of the whole dataset. So, in this version, the iterative process is performed as follows: in the K-th iteration, the algorithm alternates between completing the unlabelled instance by the previously learnt model $\psi^{(K)}$ (E-step) and learning a new model $\psi^{(K+1)}$ by using a learning algorithm with the whole dataset, both labelled and completed instances (M-step). In the semi-supervised learning research community, the input initial parameter $\psi^{(K=0)}$ of

the EM Algorithm is usually learnt from the labelled subset D_L. Hence, we will continue to use this modus operandi in this version of the algorithm. Note that our version of the EM algorithm is closer to the Bayesian structural EM algorithm proposed in [19] rather than the original formulation of the algorithm [15]. Note that, in the case of the MDnB classifier, it is just a parametric search since it has a fixed structure.

Using Algorithm 2, all the supervised learning approaches proposed in the previous section can be straightforwardly used in this semi-supervised scenario. The learning algorithm is used in the M-STEP, where it learns a model using labelled and unlabelled data that have been previously labelled in the E-STEP. So, applying our adaptation of the EM Algorithm, we have extended the multi-dimensional Bayesian network classifiers to the semi-supervised learning framework.

7. Experimentation

7.1. Artificial Experimentation

Before solving the ASOMO problem, we tested our proposed algorithms over a set of designed artificial datasets as commonly carried out in the machine learning research community. A detailed report of these artificial experiments can be found in the following website[3].

The major conclusions extracted from this experimentation can be summarised as follows:

1. As happens in the uni-dimensional framework [10], when using the real structure to semi-supervisely learnt multi-dimensional classifiers, the unlabelled data always helps.
2. There is a tendency to achieve better classifiers in terms of joint accuracy in the semi-supervised framework when the used multi-dimensional algorithm can reach the generative structure.
3. In the uni-dimensional approaches, performance degradation occurs in the semi-supervised framework. This is probably due to the fact that the uni-dimensional approaches are not able to match the actual multi-dimensional structure of the problems.
4. Although there are small differences between the uni-dimensional and the multi-dimensional approaches in the supervised framework (only the MD J/K reports statistical differences), in the semi-supervised framework these differences grow larger (except for the case of the MDTAN learning algorithm, the rest of the multi-dimensional approaches report statistical differences).
5. In the semi-supervised framework, clearly the multi-dimensional classifiers outperform the uni-dimensional techniques, with the exception of the MDTAN classifier.
6. The MDnB learning algorithm [47] is very specific, it obtains very good results when dealing with problems with an underlying MDnB structure, but when the generative models are more complex, its rigid structure makes the algorithm lead to very suboptimal solutions.
7. The MDTAN algorithm [47] also shows very poor performances in the semi-supervised framework.
8. The MD J/K learning algorithms have great flexibility to capture different types of complex structures, which results in an improvement in terms of joint accuracy in the semi-supervised framework.

In brief, these artificial experiments show that not only the multi-dimensional approaches statistically outperform the uni-dimensional approaches in the supervised framework when dealing with multi-dimensional problems, but also more accurate classifiers can be found using the semi-supervised learning framework.

[3] http://www.sc.ehu.es/ccwbayes/members/jonathan/home/News_and_Notables/Entries/2010/11/30_Artificial_Experiments_2010.html

7.2. Solving the ASOMO SA problem

Two different series of experiments with the ASOMO corpus were performed: The first series (Section 7.2.1) shows the comparison between the ASOMO features and three different state-of-the-art feature sets. In the second series (Section 7.2.2), a multi-dimensional classification solution for the ASOMO problem is shown and discussed in both supervised and semi-supervised scenarios. By means of these experiments in a real SA problem, we would like to shed some light on the truthfulness of the following hypotheses:

1. The choice of the feature set is a key matter when dealing with the exposed problem, and in extension, with the SA problems.
2. The uni-dimensional models obtained with the common approaches of the SA domain yield suboptimal solutions to the ASOMO problem.
3. The explicit use of the relationships between the class variables in this real-world problem can be beneficial to improve their recognition rates, i.e. multi-dimensional techniques are able to outperform the most common uni-dimensional techniques.
4. Multi-dimensional techniques can work with unlabelled data in order to improve the classification rates in this context.

7.2.1. Comparison between different feature sets in both uni-dimensional and multi-dimensional learning scenarios

By means of this experiment, we evaluate the new feature set proposed in Section 2 (the ASOMO features) with the most commonly used in the related literature. If we only use them to test the multi-dimensional Bayesian network classifiers in this real dataset, it is difficult to assess their efficacy without using other viable baseline methods. For this reason, we also use three different commonly used feature sets in order to perform a benchmark comparison: *unigrams*, *unigrams + bigrams* and *PoS*. In order to avoid computation burden, we limited consideration to (1) the 357 unigrams that appear at least five times in our 150-post corpus, (2) the 563 unigrams and bigrams occurring at least five times, and (3) the PoS to 766 features. No stemming or stoplists were used. These benchmark feature sets have been constructed using the TagHelper tool [42]. Finally, since the ASOMO features are continuous, they were discretised into three values using equal frequency discretisation in order to apply the algorithms proposed in this paper.

To assess the efficacy and efficiency of the exposed multi-dimensional Bayesian network classifiers in this problem, we conducted a comparison in the supervised framework between the presented multi-dimensional classifiers and the two different uni-dimensional attempts to adapt single-class classifiers to multi-dimensional problems: (1) Develop multiple uni-dimensional classifiers and (2) construct a Cartesian product class variable. In order to perform such comparison, we use naive Bayes classifiers in the uni-dimensional attempts (one per each class variable in the first uni-dimensional approach, and another one for the compound class of the second) and a MDnB classifier in the multi-dimensional solution. The parameters of both types of models are calculated by MLE corrected with Laplace smoothing and the forbidden configurations of the vector of class variables have been taken into account in the learning process. Note that this comparison is performed in the supervised framework, so the $2,392$ unlabelled posts were not used in this experiment.

The classification accuracies, which have been estimated via 20 runs of 5-fold non-stratified cross validation (20×5cv) [41], are shown in Table 6. The results of using the four different feature sets (unigrams, unigrams + bigrams, PoS, and the ASOMO features) in conjunction with

the three different learning approaches (multiple uni-dimensional, Cartesian class variable, and multi-dimensional classifiers) in a supervised framework are shown. The first eight rows of Table 6 correspond to the two uni-dimensional approaches while the last four correspond to the multi-dimensional approach. In addition to the estimation of the joint accuracy, the accuracies per each class variable are also shown.

For each column, the best estimated accuracy rate is highlighted in **bold** for each single class variable and the joint accuracy. Moreover, statistical hypothesis tests (Student's t-test with $\alpha = 0.05$) have been performed in order to determine if there are significant statistical differences between the tested accuracies. For each column, the symbol '†' is used to mark the accuracies that are statistically outperformed by the highlighted best estimated accuracy (in **bold**). The symbol ‡' corresponds to the accuracies that, despite not being significantly worse, show a p-value on the significant borderline ($0.05 < p$-value ≤ 0.1). Besides, the CPU time spent on learning a classification model with the 150 posts (for each approach and feature set) is shown in the results.

Based on the results of Table 6, different conclusions can be extracted:

1. With respect to the feature set comparison, the ASOMO feature set significantly outperforms the n-grams, not only in terms of joint accuracy, but also in terms of accuracy of two (out of the three) dimensions, i.e. in $\mathrm{Will\ to\ Influence}$ and in $\mathrm{Subjectivity}$. Also, as stated in [34], n-grams information is the most effective feature set in the task of $\mathrm{Sentiment\ Polarity}$ classification. However, the unigrams + bigrams feature set outperforms unigrams instead of the opposite as reported in [34]. Finally, the PoS approach obtains very low performance.
2. According to the comparison between the uni-dimensional and multi-dimensional approaches, the best joint accuracy is given by the MDnB model when using the ASOMO features. It significantly outperforms both the multi-dimensional approaches with the state-of-the-art feature sets, and the uni-dimensional approaches. Looking at the class variables in isolation, the uni-dimensional approaches only outperform the multi-dimensional approach in the case of the $\mathrm{Will\ to\ Influence}$ target dimension.
3. An analysis of the CPU computational times show that ,as expected, learning one classifier per each class variable is the most time-consuming. Next, multi-dimensional Bayesian network classifiers are found. The least-time consuming approach is the uni-dimensional learning process that uses a compound class. Moreover, the number of features is also an important issue with respect to the computational time. The ASOMO feature set, which has only 14 attributes, is, by far, the least time-consuming of the four different types of feature set.

		Feature set	#feat.	Will to Influence Acc	Sentiment P. Acc	Subjectivity Acc	JOINT Acc	TIME
UNI-DIMEN.	Multiple classif.	unigrams	357	†51.83 ± 2.79	33.00 ± 1.82	†78.90 ± 2.16	†11.70 ± 1.68	1,011 ms
		uni. + bigrams	563	†47.17 ± 2.47	32.90 ± 1.96	†79.47 ± 2.75	†9.80 ± 1.47	1,470 ms
		PoS	766	†47.57 ± 2.52	†30.80 ± 2.75	†66.33 ± 2.96	†9.43 ± 1.65	1,755 ms
		ASOMO feat.	14	**55.07 ± 1.32**	†26.07 ± 2.09	†82.17 ± 1.35	†10.37 ± 2.30	110 ms
	Cartesian class	unigrams	357	N/A	N/A	N/A	†9.90 ± 2.35	32 ms
		uni. + bigrams	563	N/A	N/A	N/A	†8.87 ± 2.05	74 ms
		PoS	766	N/A	N/A	N/A	†10.33 ± 2.11	106 ms
		ASOMO feat.	14	N/A	N/A	N/A	‡13.70 ± 2.60	17 ms
MULTI-DIMEN.	MDnB	unigrams	357	†41.63 ± 3.74	†31.00 ± 2.29	†75.30 ± 2.20	†10.23 ± 1.92	47 ms
		uni. + bigrams	563	†38.03 ± 3.33	**33.40 ± 2.37**	†74.27 ± 2.71	†9.63 ± 2.05	107 ms
		PoS	766	†39.50 ± 3.32	†30.53 ± 2.22	†76.40 ± 2.47	†9.86 ± 1.93	122 ms
		ASOMO feat.	14	†53.23 ± 1.62	†30.87 ± 2.52	**83.53 ± 0.69**	**14.97 ± 1.94**	17 ms

Table 6: Estimated accuracy values on the ASOMO dataset using three different types of feature sets in both uni and multi-dimensional scenarios (20×5cv).

		Feature set	#feat.	Will to Influence MAE	Sentiment P. MAE	Subjectivity MAE	JMAE
UNI-DIMEN.	Multiple classif.	unigrams	357	†0.632 ± 0.040	†1.048 ± 0.042	†0.211 ± 0.017	†0.684 ± 0.025
		uni. + bigrams	563	†0.700 ± 0.043	**0.956 ± 0.043**	†0.196 ± 0.014	†0.669 ± 0.023
		PoS	766	†0.878 ± 0.063	†1.147 ± 0.052	†0.339 ± 0.043	†0.918 ± 0.054
		ASOMO feat.	14	0.563 ± 0.014	†1.036 ± 0.036	†0.173 ± 0.011	†0.620 ± 0.014
	Cartesian class	unigrams	357	N/A	N/A	N/A	†0.650 ± 0.026
		uni. + bigrams	563	N/A	N/A	N/A	†0.680 ± 0.030
		PoS	766	N/A	N/A	N/A	†0.757 ± 0.040
		ASOMO feat.	14	N/A	N/A	N/A	†0.640 ± 0.060
MULTI-DIMEN.	MDnB	unigrams	357	†0.788 ± 0.049	†1.104 ± 0.039	†0.247 ± 0.026	†0.789 ± 0.031
		uni. + bigrams	563	†0.852 ± 0.052	†1.107 ± 0.042	†0.263 ± 0.040	†0.824 ± 0.049
		PoS	766	†0.728 ± 0.044	†1.037 ± 0.040	†0.240 ± 0.020	†0.742 ± 0.029
		ASOMO feat.	14	**0.559 ± 0.029**	†1.019 ± 0.027	**0.167 ± 0.009**	**0.608 ± 0.016**

Table 7: Estimated mean absolute error rates on the ASOMO dataset using three different types of feature sets in both uni and multi-dimensional scenarios (20×5cv).

However, in this problem, errors are not simply present or absent, their magnitude can be computed, e.g. it is not the same to misclassify a negative post as having a very negative sentiment or misclassify it with a very positive sentiment. For that reason, in addition to the accuracy term, we also estimate the numeric error of each classifier. Note that the values of the three class variables can be trivially translated into ordinal values without changing their meaning. Therefore, using this approach, the previous example could be exposed as: It is not the same misclassify a post, which has its sentiment equal to 2, with a 1 or misclassify it with a 5.

In order to estimate the numeric error, we use the mean absolute error (MAE) term [49], which is a measure broadly used in evaluating numeric prediction. It is defined as the measure that averages the magnitude of the individual errors without taking their sign into account . It is given by the following formula:

$$MAE\epsilon_j(\psi) = \frac{\sum_{i=1}^{N} |\psi_j(\mathbf{x}^{(i)}) - c_j^{(i)}|}{N}$$

where $\psi_j(\mathbf{x}^{(i)})$ is the value of the class variable C_j resulting from the classification of the instance $\mathbf{x}^{(i)}$ using the classifier ψ_j and $c_j^{(i)}$ is the actual class value in that instance. N is the number of instances in the test set. Note that the resulting error varies between 0 and $(|C_j| - 1)$, where $|C_j|$ is the cardinality of the class variable C_j.

In a similar way to the accuracy, we also compute a joint measure for simultaneously characterising this error in all the class variables. Due to this, we estimate the joint MAE (JMAE) for each learning algorithm. It is the sum of the normalised value of the MAE term in each class variable.

$$JMAE\epsilon(\psi) = \sum_{j=1}^{m} \frac{1}{|C_j| - 1} MAE\epsilon_j(\psi)$$

Note that the JMAE term varies between 0 and m, being m the number of class variables.

The MAE values of the exposed experimentation setup, which have also been estimated via 20 runs of 5-fold non-stratified cross validation [41], are shown in Table 7. It has the same shape as Table 6, i.e. each row represents each learning algorithm with a specific feature set, and each column represents each class variable and the JMAE value. The best estimated error per classifier is also highlighted in **bold** and Student's t-tests ($\alpha = 0.05$) have been performed in order to study the significance of estimated differences. Table 7 reports conclusions similar to the ones extracted with the accuracy term:

1. The feature set comparison reports the same conclusions to those obtained with the accuracy: The ASOMO feature set significantly outperforms the n-grams and PoS, not only in terms of joint accuracy, but also in terms of two (out of the three) dimensions.
2. The best joint accuracy is given by the multi-dimensional approach which uses the ASOMO feature set and it significantly outperforms both the multi-dimensional approaches with the state-of-the-art feature sets, and the uni-dimensional approaches. With respect to the class variables in isolation, the only case in which the uni-dimensional approaches outperform the multi-dimensional approach is in the $\mathrm{Sentiment\ Polarity}$ target dimension.

In brief and regarding the feature set, for this specific problem, we strongly recommend the use of the ASOMO features, not only because of their performance, but also for their lower learning times. The results also show that the multi-dimensional classification approach to SA is

a novel attractive point of view that needs to be taken into account due to the fact that it could lead to better results in terms of accuracy as well as in MAE. In addition, learning a multi-dimensional classifier is faster than learning different classifiers for each dimension. Although the reported times are not a problem in the current supervised learning settings, in the semi-supervised framework, where the computation time increases dramatically with the number of instances, the learning process could be intractable.

7.2.2. Experiments with the ASOMO corpus in the supervised and semi-supervised learning frameworks

With the knowledge that the ASOMO features can lead us to better classification rates in this problem, we evaluated their performance in both supervised and semi-supervised frameworks. With this experiment we want to determine if the use of unlabelled examples when learning a classifier can lead to better solutions to the ASOMO problem. In order to do so, the following experiment was performed: The ASOMO dataset has been used to learn three different (uni-dimensional) Bayesian network classifiers and three different sub-families of multi-dimensional classifiers in both frameworks.

For uni-dimensional classification, naive Bayes classifier (*nB*), tree-augmented Bayesian network classifier (*TAN*) [20] and a 2-dependence Bayesian classifier (2 DB) [43] have been chosen. The uni-dimensional approach selected for these experiments is that which consists of splitting the problem into three different uni-dimensional problems (this is because it is more common in the state-of-the-art solutions to solve different problems rather than create an artificial class variable by means of the Cartesian product). From the multi-dimensional aspect, *MDnB*, *MDTAN*, *MD* 1/1 and *MD* 2/K (with $K = 2, 3, 4$) structures have been selected. *MD* 1/1 is included as an algorithm able to learn MDTAN structures due to the poor performance shown by the MDTAN learning algorithm [47] in the artificial experiments. Although both multi-dimensional earning approaches learn MDTAN structures, each learning algorithm follows a different path to come to that end. While the MD 1/1 uses a filter approach, the MDTAN learning algorithm follows a wrapper scheme.

The supervised learning procedure only uses the labelled dataset (consisting of 150 documents), whilst the semi-supervised approach uses the $2,532$ posts ($2,392$ unlabelled). Our multi-dimensional extension of the EM algorithm is used in the latter approach and it terminates after finding a local likelihood maxima or after 250 unsuccessful trials.

Finally, the performance of each model has been estimated via 20 runs of 5-fold non-stratified cross validation. Due to fact that, in semi-supervised learning, the labels of the unlabelled subset of instances are unknown, only the labelled subset is divided into 5 folds to estimate the performance of the proposed approaches. So, in each iteration of the cross validation, a classifier is learnt with 4 labelled folds and the whole unlabelled subset, and then it is tested in the remaining labelled fold. This modified cross validation is illustrated in Figure 7 for the case of 3 folds. As done in the previous experiments, we use the accuracy and the MAE terms as evaluation measures.

Table 8 shows the results of applying the different uni-dimensional and multi-dimensional algorithms over the ASOMO dataset in terms of accuracy and Table 9 in terms of MAE. Both tables can be described as follows: For each classifier (row), the joint performance and the single evaluation measure (for each class variable) are shown. In order to simultaneously compare uni-dimensional with respect to multi-dimensional approaches, and supervised with respect to semi-supervised learning, the results are highlighted as follows:

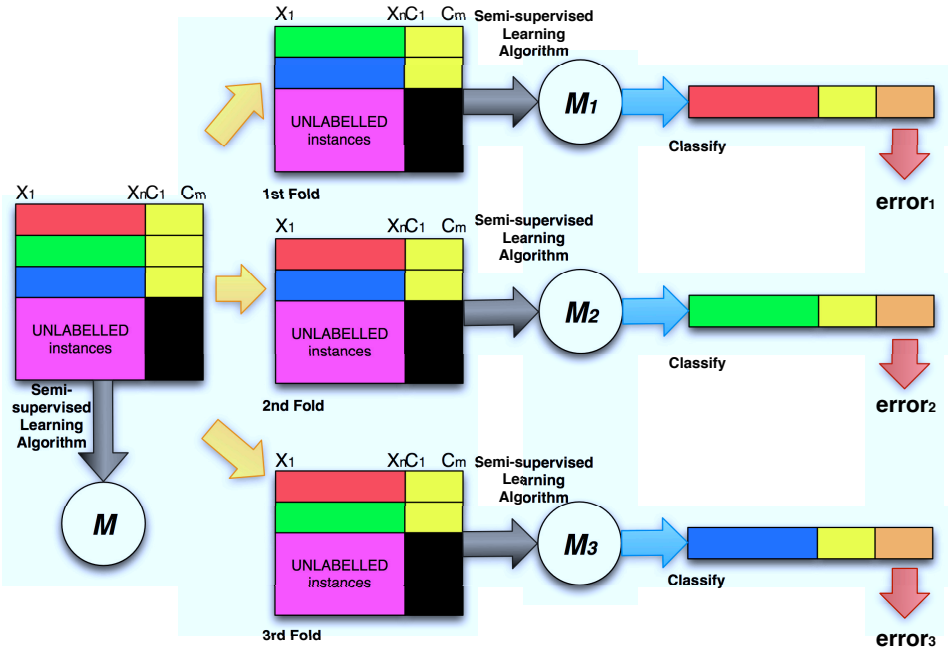


Figure 7: Applying 3 fold cross validation to a dataset with labelled and unlabelled instances.

1. In order to compare the supervised and the semi-supervised frameworks, for each type of classifier and accuracy measure (class variable and joint performance), we have highlighted the best single value and joint performance in **bold** (analysed per row). Note that, in the case of the accuracy term (Table 8), the highlighted values are the greatest values, while in the case of the MAE (Table 9) they are the lowest. Pairwise statistical hypothesis tests (Student's t-test with $\alpha = 0.05$) have been performed in order to determine if there are significant statistical differences between the values of the tested techniques. We use the symbol '†' to mark the values that are statistically outperformed by the highlighted best estimated measure (in **bold**).
2. To compare the performance between the uni-dimensional and the multi-dimensional approaches, the best accuracy per class variable, as well as the joint performance, has been highlighted in *italics*. For each column, statistical hypothesis tests (Student's t-test with $\alpha = 0.05$) have been performed in order to determine if there are significant statistical differences. The symbol '‡' is used to mark the values that are statistically worse than the best estimated value (in *italics*).

Several conclusions can be extracted from the supervised and semi-supervised comparison in Tables 8 and 9 (analysed per row):

1. The uni-dimensional models tend to perform worse when they are learnt using the semi-supervised learning framework. This could be due to the fact that incorrect models tend to lead to performance degradation in the semi-supervised framework due to the fact that they

are not able to match the underlying generative structure [10]. This phenomenon occurs in both evaluation measures.

2. As occurs in the artificial experiments, the MDTAN approach [47] tends to behave more similarly to the uni-dimensional approaches rather than to the multi-dimensional approaches.
3. With respect to the accuracy measure, in the multi-dimensional scenario, the Will to Influence class variable tends to degrade its performance in the semi-supervised scenario, whilst Sentiment Polarity and Subjectivity tend to achieve better single accuracies. In the uni-dimensional approach, the opposite happens.
4. The MAE results show that, unlike what happens in the uni-dimensional framework where the semi-supervised degradation is significant, in the multi-dimensional scenario similar results are reported for supervised and semi-supervised learning. However, there are cases in which the semi-supervised learning algorithms obtain better results and in one case there is a significant statistical gain.
5. The MDnB method in its semi-supervised framework is the best solution for the ASOMO problem. In terms of accuracy, it obtains statistically significant better results in joint accuracy and in the Sentiment Polarity target variable, as well as better results in the other two variables. With respect to the MAE, MDnB obtains statistically significant better results in joint accuracy and in the Subjectivity dimension, as well as better results in the other dimensions.

Regarding the comparison of the uni-dimensional and multi-dimensional approaches (for each column), the following comments can be extracted:

1. With the exception of Will to Influence, the single class variables tend to achieve better accuracies in the multi-dimensional approaches.
2. The MAE terms show similar results to those found with the accuracy evaluation measure. However, in this case, the exception is provided by the estimated MAE obtained in the Sentiment Polarity dimension in the semi-supervised version of the uni-dimensional TAN algorithm.
3. The multi-dimensional classification approach statistically outperforms the uni-dimensional framework in terms of joint accuracy.
4. MDnB is also the best technique in terms of global performance, i.e. in accuracy and in MAE metrics.

In conclusion, we show that the proposed semi-supervised multi-dimensional formulation designs a novel perspective for this kind of SA problems, opening new ways to deal with this domain. In addition, it can also be seen that the explicit use and the representation of the relationships between different class variables have successfully solved the ASOMO problem, where the MDnB in a semi-supervised framework is the best solution. One explanation for the surprising success of the MDnB could be the use of the knowledge of the experts in engineering the features, as stated in [24]: When applying rational criteria in determining the predictive features of a problem, the resulting features are usually probabilistically independent given the class variables. This characteristic favours the learning scheme provided by the MDnB algorithm. Moreover, this is crucial in the success obtained in the semi-supervised framework due to the fact that it matches the actual underlying domain structure [10].

Classif.	LABELLED DATA (Supervised Learning)				LABELLED + UNLABELLED DATA (Semi-supervised Learning)			
	W. Influence Acc	Sentiment P. Acc	Subjectivity Acc	JOINT Acc	W. Influence Acc	Sentiment P. Acc	Subjectivity Acc	JOINT Acc
nB	††‡55.07 ± 1.32	**‡26.07 ± 2.09**	**‡82.17 ± 1.35**	**‡10.37 ± 2.30**	***56.70 ± 2.85***	††‡21.97 ± 3.11	††‡58.07 ± 3.78	††‡7.43 ± 2.27
TAN	**‡52.93 ± 3.23**	††‡26.30 ± 2.16	**‡80.10 ± 1.46**	‡9.47 ± 1.29	††‡48.77 ± 3.44	**‡29.00 ± 3.69**	††‡75.10 ± 3.42	**‡9.87 ± 1.79**
2DB	***57.13 ± 1.85***	**‡27.43 ± 2.82**	**‡82.30 ± 1.57**	**‡13.17 ± 1.59**	††‡54.13 ± 3.37	††‡24.80 ± 2.95	††‡61.30 ± 4.83	††‡8.60 ± 2.08
MDnB	‡53.23 ± 1.62	*†30.87 ± 2.52*	*83.53 ± 0.69*	†14.97 ± 1.94	**‡54.00 ± 2.08**	***32.27 ± 1.41***	***83.63 ± 0.55***	***16.83 ± 1.14***
MDTAN	**‡52.60 ± 3.48**	††‡27.13 ± 2.43	‡82.57 ± 1.72	**‡12.80 ± 2.38**	††‡31.10 ± 4.37	**‡29.43 ± 2.48**	**‡82.60 ± 1.37**	††‡8.97 ± 1.70
MD 1/1	**56.67 ± 2.93**	**29.97 ± 3.75**	**78.17 ± 2.59**	**15.63 ± 2.00**	††‡53.27 ± 2.78	‡29.17 ± 2.54	‡77.90 ± 2.20	‡15.33 ± 1.37
MD 2/2	**56.60 ± 3.63**	**‡28.93 ± 2.36**	**‡77.47 ± 2.24**	15.17 ± 1.99	††‡53.30 ± 2.08	‡28.77 ± 2.37	‡76.93 ± 2.97	**‡15.50 ± 0.94**
MD 2/3	**56.70 ± 2.87**	29.87 ± 3.20	**‡77.03 ± 1.41**	*15.90 ± 2.67*	††‡52.77 ± 1.53	**‡30.77 ± 2.25**	‡77.90 ± 2.20	**16.63 ± 1.32**
MD 2/4	**56.97 ± 2.11**	‡28.53 ± 3.13	**‡76.87 ± 2.39**	**15.57 ± 2.41**	††‡52.47 ± 2.05	**29.30 ± 3.02**	‡75.27 ± 3.24	‡15.43 ± 1.28

Table 8: Accuracies on the ASOMO dataset with the ASOMO features in the supervised and the semi-supervised learning frameworks (20×5cv).

Classif.	LABELLED DATA (Supervised Learning)				LABELLED + UNLABELLED DATA (Semi-supervised Learning)			
	W. Influence MAE	Senti. P. MAE	Subject. MAE	JMAE	W. Influence MAE	Senti. P. MAE	Subject. MAE	JMAE
nB	**‡0.563 ± 0.014**	**1.036 ± 0.036**	**0.173 ± 0.011**	**‡0.620 ± 0.014**	††‡0.613 ± 0.040	††‡1.209 ± 0.069	††‡0.421 ± 0.036	††‡0.986 ± 0.041
TAN	**‡0.614 ± 0.041**	††‡1.044 ± 0.043	**‡0.204 ± 0.015**	**‡0.670 ± 0.029**	††‡0.664 ± 0.055	***0.991 ± 0.041***	††‡0.225 ± 0.038	††‡0.718 ± 0.039
2DB	**‡0.553 ± 0.028**	**1.035 ± 0.053**	**0.171 ± 0.009**	**0.614 ± 0.018**	††‡0.636 ± 0.041	††‡1.114 ± 0.082	††‡0.387 ± 0.048	††‡0.885 ± 0.056
MDnB	‡0.559 ± 0.029	*1.019 ± 0.027*	*†0.167 ± 0.009*	*†0.608 ± 0.016*	***0.549 ± 0.019***	**1.002 ± 0.028**	***0.612 ± 0.005***	***0.596 ± 0.007***
MDTAN	**‡0.567 ± 0.045**	**‡1.101 ± 0.052**	‡0.180 ± 0.015	**‡0.644 ± 0.026**	††‡0.878 ± 0.075	‡1.102 ± 0.028	**‡0.172 ± 0.011**	††‡0.750 ± 0.027
MD 1/1	**0.529 ± 0.034**	**‡1.056 ± 0.032**	‡0.219 ± 0.022	**‡0.659 ± 0.029**	†0.556 ± 0.031	‡1.061 ± 0.054	**‡0.216 ± 0.018**	‡0.666 ± 0.015
MD 2/2	***0.525 ± 0.033***	**‡1.057 ± 0.054**	**‡0.222 ± 0.021**	**‡0.660 ± 0.034**	†0.560 ± 0.040	‡1.075 ± 0.050	‡0.227 ± 0.023	††‡0.682 ± 0.022
MD 2/3	**0.531 ± 0.032**	**‡1.061 ± 0.032**	‡0.237 ± 0.034	‡0.679 ± 0.037	*†0.549 ± 0.016*	‡1.048 ± 0.049	**‡0.223 ± 0.033**	**‡0.678 ± 0.033**
MD 2/4	**0.529 ± 0.041**	**‡1.069 ± 0.058**	‡0.227 ± 0.020	**‡0.670 ± 0.031**	††‡0.579 ± 0.038	‡1.047 ± 0.038	**‡0.225 ± 0.023**	‡0.680 ± 0.021

Table 9: Mean absolute error rates on the ASOMO dataset with the ASOMO features in the supervised and the semi-supervised learning frameworks (20×5cv).

8. Conclusions and future work

In this paper, we solve a real-world multi-dimensional SA problem. This real problem consists of characterising the attitude of a customer when he writes a post about a particular topic in a specific forum through three differently related dimensions: $\mathrm{Will\ to\ Influence}$, $\mathrm{Sentiment\ Polarity}$ and $\mathrm{Subjectivity}$.

Due to the fact that it has three different target variables, the SA (uni-dimensional) state-of-the-art classification techniques obtain suboptimal solutions. For that reason, we propose the use of multi-dimensional Bayesian network classifiers as a novel methodological tool to take advantage of these existing relations between these target variables. Besides, in order to avoid the arduous and time-consuming task of labelling examples in this field, we present these multi-dimensional techniques in both supervised and semi-supervised learning frameworks.

Experimental results of applying the proposed battery of multi-dimensional learning algorithms to a corpus consisting of $2,542$ posts (150 manually labelled and $2,392$ unlabelled) show that: (1) the uni-dimensional approaches cannot capture the multi-dimensional underlying nature of this problem, (2) engineering a suitable feature set is a key factor for obtaining better solutions, (3) more accurate classifiers can be found using the multi-dimensional approaches, (4) the use of large amounts of unlabelled data in a semi-supervised framework can be beneficial to improve the recognition rates, and (5) the MDnB classifier in a semi-supervised framework is the best solution for this problem because it matches the actual underlying domain structure [24][10].

The proposed multi-dimensional methodology can be improved or extended in several ways. For instance, in the ASOMO multi-dimensional problem, the values of all class variables are missing in each sample of the unlabelled subset. However, by means of the EM algorithm, the learning algorithms can be easily generalised to the situation where not all the class variables are missing in all the samples of the unlabelled data subset.

Besides, we are concerned about the scalability of the multi-dimensional Bayesian network classifiers in the semi-supervised framework. The computational burden is not a problem when dealing with these datasets, but it could happen when the number of variables increases. This could open a line in researching feature subset selection techniques for multi-dimensional classification.

Regarding the application of multi-dimensional classification to the SA domain, this work can be extended in a number of different ways:

- The proposed multi-dimensional Bayesian network classifiers can be directly applied to Affect Analysis. This area is concerned with the analysis of text containing emotions and it is associated with SA [1]. However, Affect Analysis tries to extract a large number of potential emotions, e.g. happiness, sadness, anger, hate, violence, excitement, fear, etc, instead of just looking at the polarity of the text. Additionally, in the case of Affect Analysis, the emotions are not mutually exclusive and certain emotions may be correlated. So, this can easily be viewed as a multi-label classification problem, a type of problem in which multi-dimensional Bayesian network classifiers have reported good results in the recent past [4].

- Within SA, the same corpus can be used to deal with different target dimensions. This could open different research lines in adding more target variables in the same classification task so as to take advantage of these existing relationships, engineering a suitable feature set for working with several dimensions, etc. For instance, in the works where the need to predict both the Sentiment Polarity and the Subjectivity has been noticed [17].

Acknowledgments

This work has been partially supported by the Saiotek, Etortek and Research Groups 2007-2012 (IT-242-07) programs (Basque Government), TIN2008-06815-C02-01, TIN2010-14931, Consolider Ingenio 2010-CSD2007-00018 projects and MEC-FPU grant AP2008-00766 (Spanish Ministry of Science and Innovation), and COMBIOMED network in computational biomedicine (Carlos III Health Institute).

References

[1] A. Abbasi, H. Chen, S. Thoms, and T. Fu. Affect analysis of web forums and blogs using correlation ensembles. *IEEE Transactions on Knowledge and Data Engineering*, 20(9):1168–1180, 2008.

[2] S. Argamon, C. Whitelaw, P. Chase, S. Raj Hota, N. Garg, and S. Levitan. Stylistic text classification using functional lexical features. *Journal of the American Society for Information Science and Technology*, 58(6):802–822, 2007.

[3] J. Atserias, B. Casas, E. Comelles, M. Gonzalez, L. Padro, and M. Padro. Freeling 1.3: Syntactic and semantic services in an open-source NLP library. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'06)*, pages 48–55, 2006.

[4] C. Bielza, G. Li, and P. Larrañaga. Multi-dimensional classification with Bayesian networks. Technical Report UPM-FI/DIA/2010-1, Departmento de Inteligencia Artificial, Universidad Politécnica de Madrid, Madrid, Spain, 2010.

[5] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2006.

[6] R. Blanco. *Learning Bayesian Networks from Data with Factorisation and Classification Purposes. Applications in Biomedicine*. PhD thesis, University of the Basque Country, 2005.

[7] X. Carreras, I. Chao, L. Padro, and M. Padro. An open-source suite of language analyzers. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*, volume 10, pages 239–242, 2006.

[8] O. Chapelle, B. Scholkopf, and A. Zien. *Semi-supervised learning*. The MIT Press, 2006.

[9] C. I. Chow, S. Member, and C. N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14:462–467, 1968.

[10] I. Cohen. *Semisupervised Learning of Classifiers with Application to Human-Computer Interaction*. PhD thesis, University of Illinois at Urbana-Champaign, 2003.

[11] I. Cohen, F. G. Cozman, and A. Bronstein. The effect of unlabeled data on generative classifiers, with application to model selection. In *Proceedings of the SPIE 93 Conference on Geometric Methods in Computer Vision*, 2002.

[12] I. Cohen, F. G. Cozman, N. Sebe, M. C. Cirelo, and T. S. Huang. Semisupervised learning of classifiers: Theory, algorithms and their application to human-computer interaction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(12):1553–1567, 2004.

[13] G. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309–347, 1992.

[14] P. R. de Waal and L. C. van der Gaag. Inference and learning multi-dimensional Bayesian network classifiers. *Lecture Notes in Artificial Intelligence*, 4724:501–511, 2007.

[15] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[16] Boiy E. and Moens M. F. A machine learning approach to sentiment analysis in multilingual web text. *Information Retrieval*, 12(5):526–558, 2009.

[17] A. Esuli and F. Sebastiani. Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC06)*, pages 417–422, 2006.

[18] G. Forman. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning*, 3:1289–1305, 2003.

[19] N. Friedman. The Bayesian structural EM algorithm. In *Proceedings of the 14th Conference on Incertainty in Artificial Intelligence*, pages 129–138. Morgan Kaufmann Publishers, 1998.

[20] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2-3):131–163, 1997.

[21] J. H. George, R. Kohavi, and K. Pfleger. Irrelevant features and the subset selection problem. In *Machine Learning, Proceedings of the Eleventh International Conference*, pages 121–129, 1994.

[22] H. Hartley. Maximum likelihood estimation from incomplete data. *Biometrics*, 14:174–194, 1958.

[23] V. Hatzivassiloglou and J. M. Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th conference on Computational linguistics*, pages 299–305, 2000.

[24] R. Kohavi. *Wrappers for Performance Enhancement and Oblivious Decision Graphs*. PhD thesis, Stanford University, 1995.

[25] J. B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society*, 7(1):48–50, 1956.

[26] S. Kullback. *Information Theory and Statistics*. Wiley, 1959.

[27] P. Langley, W. Iba, and K. Thompson. An analysis of Bayesian classifiers. In *Proceedings of the 10th National Conference on Artificial Intelligence*, pages 223–228, 1992.

[28] P. Larrañaga, J. A. Lozano, J. M. Peña, and I. Inza. *Special Issue on Probabilistic Graphical Models for Classification*, volume 59(3). Machine Learning, 2005.

[29] B. Liu. Sentiment analysis and subjectivity. In N. Indurkhya and F. J. Damerau, editors, *Handbook of Natural Language Processing*. Chapmand & Hall, second edition, 2010.

[30] G. J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley-Interscience, 1997.

[31] V. Ng, S. Dasgupta, and S. M. N. Arifin. Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews. In *Proceedings of the COLING/ACL Conference*, pages 611–618, 2006.

[32] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2):103–134, 2000.

[33] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, 2008.

[34] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'02)*, pages 79–86, 2002.

[35] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers Inc., 1988.

[36] M. Qazi, G. Fung, S. Krishnan, R. Rosales, H. Steck, R. B. Rao, D. Poldermans, and D. Chandrasekaran. Automated heart wall motion abnormality detection from ultrasound images using Bayesian networks. In *IJCAI'07: Proceedings of the 20th International Joint Conference on Artifical Intelligence*, pages 519–525. Morgan Kaufmann Publishers Inc., 2007.

[37] E. Riloff and J. Wiebe. Learning extraction patterns for subjective expressions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'03)*, pages 105–112, 2003.

[38] E. Riloff, J. Wiebe, and T. Wilson. Learning subjective nouns using extraction pattern bootstrapping. In *Proceedings of the 7th Conference on Natural Language Learning*, pages 25–32, 2003.

[39] J. D. Rodríguez and J. A. Lozano. Multi-objective learning of multi-dimensional Bayesian classifiers. In *Eighth Conference on Hybrid Intelligent Systems (HIS'08)*, pages 501–506, September 2008.

[40] J. D. Rodríguez and J. A. Lozano. Learning Bayesian network classifiers for multi-dimensional supervised classification problems by means of a multi-objective approach. Technical Report EHU-KZAA-TR-3-2010, Department of Computer Science and Artificial Intelligence, University of the Basque Country, San Sebastián, Spain, 2010.

[41] J. D. Rodriguez, A. Perez, and Lozano J. A. Sensitivity analysis of k-fold cross-validation in prediction error estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(3):569–574, 2010.

[42] C. Rose, Y.C. Wang, Y. Cui, J. Arguello, K. Stegmann, A. Weinberger, and F. Fischer. Analyzing collaborative learning processes automatically: Exploiting the advances of computational linguistics in computer-supported collaborative learning. *International Journal of Computer-Supported Collaborative Learning*, 3(3), 2007.

[43] M. Sahami. Learning limited dependence Bayesian classifiers. In AAAI Press, editor, *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*, pages 335–338, 1996.

[44] B. Snyder and R. Barzilay. Multiple aspect ranking using the good grief algorithm. In *Proceedings of the Joint Human Language Technology/North American Chapter of the ACL Conference (HLT-NAACL)*, pages 300–307, 2007.

[45] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society B*, 36(1):111–147, 1974.

[46] O. Tsur, D. Davidiv, and A. Rappoport. ICWSM A great catchy name: Semi-supervised recognition of sarcastic sentences in product reviews. In *Proceedings of the International AAAI Conference on Weblogs and Social Media - ICWSM10*, 2010.

[47] L. C. van der Gaag and P. R. de Waal. Multi-dimensional Bayesian network classifiers. In *Proceedings of the Third european workshop in probabilistic graphical models*, pages 107–114, 2006.

[48] J. Wiebe, T. Wilson, R. Bruce, N. Bell, and M. Martin. Learning subjective language. *Computational Linguistics*, 30(3):277–308, 2004.

[49] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, second edition, 2005.

[50] H. Yu and V. Hatzivassiloglou. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the Conference on Empirical Methods in Natural Language*

Processing (EMNLP), 2003.

[51] X. Zhu. Semi-supervised learning literature survey. Technical Report 1530, Department of Computer Sciences, University of Wisconsin, Madison, 2005.

[52] X. Zhu and A. B. Goldberg. *Introduction to Semi-Supervised Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2009.