Dimensionality Reduction for Machine Learning-based Argument Mining

Andrés Segura-Tinoco and **Iván Cantador**
Universidad Autónona de Madrid, Madrid, Spain
andres.segurat@uam.es, ivan.cantador@uam.es

Abstract

Recent approaches to argument mining have focused on training machine learning algorithms from annotated text corpora, utilizing as input high-dimensional linguistic feature vectors. Differently to previous work, in this paper, we preliminarily investigate the potential benefits of reducing the dimensionality of the input data. Through an empirical study, testing SVD, PCA and LDA techniques on a new argumentative corpus in Spanish for an underexplored domain (e-participation), and using a novel, rich argument model, we show positive results in terms of both computation efficiency and argumentative information extraction effectiveness, for the three major argument mining tasks: argumentative fragment detection, argument component classification, and argumentative relation recognition. On a space with dimension around 3-4% of the number of input features, the argument mining methods are able to reach 95-97% of the performance achieved by using the entire corpus, and even surpass it in some cases.

1 Introduction

Since its origins in the late 2000s, the argument mining (AM) field has witnessed significant advances on the problem of automatically extracting structured argumentative information from text corpora (Lytos et al., 2019; Lawrence and Reed, 2020), which commonly entails three tasks: the identification of argumentative fragments in an input text, the split or classification of such fragments into argument components (e.g., *claims* and *premises*), and the recognition of relations (e.g., *support* and *attack*) between pairs of argument components.

In particular, previous research has led to the development of effective approaches based on machine learning (ML) (Lippi and Torroni, 2015, 2016) with results almost equal to those obtained with more complex approaches, such as those based on deep learning. Hence, argumentative fragment detection (Mochales Palau and Moens, 2009a, 2011; Poudyal et al., 2016), argument component classification (Habernal and Gurevych, 2017; Du et al., 2017), and argument relation recognition (Du et al., 2017) have been modeled as *sequence labeling* problems, where, in general, each sentence[1] is represented as a vector of real-valued linguistic features and has associated certain label or class, e.g., *argumentative* vs. *non-argumentative*, and *claim* vs. *premise*. ML algorithms are thus trained with sets of labeled sentence vectors in order to predict the class of new sentences.

In this context, a variety of features have been considered –ranging from lexical and morphological, to structural and syntactic, and semantic and discourse features (Stab and Gurevych, 2014; Aker et al., 2017; Habernal and Gurevych, 2017)– and, in general, approaches have dealt with feature vectors of high dimensionality.

To the best of our knowledge, only a few research attempts have been made to use a subset of features (Poudyal et al., 2016; Du et al., 2017). Motivated by this fact and the increasing need for more efficient (i.e., less resource-consuming) AM model building, in this paper, instead of exploring new argument-related classification algorithms, we investigate the potential benefits of reducing the dimensionality of the input data space.

As an innovative research in the AM field, we report experiments conducted with the well known SVD (Beltrami, 1973; Stewart, 1993), PCA (Hotelling, 1933) and LDA (Fisher, 1936) dimensionality reduction techniques on a novel corpus in Spanish with electronic (online) citizen participation discussions, which represent an underexplored domain in the field.

Considering a rich argument model with several argument relations, and addressing the argumenta-

[1]The majority of feature-based AM approaches consider the sentence as the argumentative unit. However, there are models that also exploit other text fragments, such as the previous and next sentences to the current sentence (Habernal and Gurevych, 2017).

Proceedings of the 10th Workshop on Argument Mining, pages 89–99
December 7, 2023.

tive fragment detection, argument component classification, and argument relation recognition tasks, we evaluate a number of ML algorithms trained with labeled data without and with dimensionality reduction, achieving favorable results in terms of both computation efficiency and argumentative information extraction effectiveness. With around 3-4% of the number of input features, the argument mining methods are able to reach 95-97% of the performance achieved by using the entire corpus, and even surpass it in some cases.

We thus claim the following contributions of our ongoing work:

- Building a new argumentative corpus in Spanish, on an underexplored, but highly relevant domain: e-participation, and more specifically, e-participatory budgeting.
- Proposing a new argument model, which includes a variety of fine-grained types of argumentative relations.
- Developing and evaluating machine learning-based methods for the main tasks of the argument mining pipeline.
- Testing the effects of dimensionality reduction on the efficiency and effectiveness of the argument mining methods.

Moreover, we make the generated argument model, corpus, annotation tool, software code, and empirical results publicly available[2].

Before presenting our experiments (Section 5) and conclusions (Sections 6 and 7), we next survey related work on feature-based machine learning AM (Section 2), and describe the addressed case study and created corpus (Section 3) and the used dimensionality reduction techniques (Section 4).

2 Related work

In this section, we survey previous work on applying feature-based machine learning for AM. We discard deep learning approaches, since they are appropriate to very large amounts of input data[3].

Feature-based ML methods model AM tasks as sequence labeling problems. They have been proposed to separately address the argumentative fragment detection (Mochales Palau and Moens, 2009a, 2011; Poudyal et al., 2016; Kunaefi and Aritsugi, 2020; Alhamzeh et al., 2021), argument component classification (Mochales Palau and Moens, 2009a, 2011; Habernal and Gurevych, 2017; Burhan ud Din Tahir, 2017), and argument relation recognition (Du et al., 2017) tasks.

The surveyed methods consider the sentence as the argument unit, and exploit its linguistic features to classify it. Only Habernal and Gurevych (2017) also exploited feature information from the previous and next sentences to the target sentence. In all cases, however, the used features are manifold, as we will detail in Section 4.1.

From our survey, only Du et al. (2017) addressed the argument relation recognition task. This is not the case in recent word embedding-based deep learning methods, which deal with the three tasks as *sequence tagging* problems, by commonly following the BIO tagging format, e.g., (Deguchi and Yamaguchi, 2019; Mayer et al., 2020).

With respect to the used ML algorithms, published work has focused on logistic regression (Du et al., 2017; Kunaefi and Aritsugi, 2020), naive Bayes (Mochales Palau and Moens, 2009a, 2011; Burhan ud Din Tahir, 2017), maximum entropy (Mochales Palau and Moens, 2009a, 2011), decision trees (Burhan ud Din Tahir, 2017; Du et al., 2017), random forests (Poudyal et al., 2016; Burhan ud Din Tahir, 2017; Du et al., 2017; Kunaefi and Aritsugi, 2020), and support vector machines (Mochales Palau and Moens, 2009a, 2011; Poudyal et al., 2016; Burhan ud Din Tahir, 2017; Du et al., 2017; Kunaefi and Aritsugi, 2020; Alhamzeh et al., 2021).

Additionally, as done in deep learning works, the surveyed papers have focused on the traditional domains and corpora of the AM field, such as the Persuasive Student Essays corpus (Burhan ud Din Tahir, 2017; Du et al., 2017; Alhamzeh et al., 2021), the legal texts ECHR (Mochales Palau and Moens, 2009a, 2011; Poudyal et al., 2016) and AraucariaDB (Mochales Palau and Moens, 2009a, 2011) corpora, and the Web Discourse corpus (Alhamzeh et al., 2021), which are mostly in English.

To the best of our knowledge, in the AM research literature, only Poudyal et al. (2016) and Du et al. (2017) have explored feature selection using information gain, reducing the input feature vector space. In this context, a traditional pre-learning di-

[2]Data and code are available at `https://github.com/argrecsys/arg-classifier`

[3]Experimenting with some deep neural network architectures, we did not achieve better performance results than those reported in this paper with traditional machine learning algorithms.

mensionality reduction approach, such as the ones we evaluate here, has not been explored yet.

Differently from (Poudyal et al., 2016; Kunaefi and Aritsugi, 2020; Alhamzeh et al., 2021), we do not only focus on classifying a sentence as *argumentative* or *non-argumentative*, but also aim to classify the type of argumentative component of a text span, i.e., a *premise* or a *claim*, and to recognize the existence of a relation between a pair of components and its type.

Finally, motivated by the need for addressing other domains and dealing with corpora in languages distinct to English, in our work we explore a novel domain in AM and provide a new corpus in Spanish, which are described next.

3 Case study

In this section, we introduce the case study for which we have built our argumentative corpus and have implemented and evaluated the machine learning-based argument mining methods with and without dimensionality reduction techniques.

Citizen participation refers to the active involvement of citizens in influencing on public opinion and being part of democratic decision and policy making processes. It represents one of the most widespread forms of open government, and historically has been conducted through physical interactions like assemblies, meetings and working groups (Gramberger, 2001).

Nowadays, under the umbrella of e-participation (Boudjelida et al., 2016), it often occurs on the internet, via online digital tools, in which citizens' opinions and contributions are easily shared, thus generating new opportunities for communication, consultation and collaboration at a large scale (Held, 2006).

The majority of current e-participation platforms are based on web forums where citizens upload comments, forming large conversation threads. This makes the processing of the underlying debates challenging and sometimes overwhelming. Without functionalities to support critical thinking and argumentation, it is usually very difficult and time-consuming for users to achieve a well-formed view of existing problems and proposed solutions.

In our work, we focus our attention on one of such platforms, Decide Madrid[4], which is an online web portal created by the City Council of Madrid (Spain) to support its annual participatory budgets since September 2015. Every year, the city residents use the platform to freely post proposals to address issues and problems in the city, and comment and vote others' proposals. Those citizen proposals that receive a certain number of votes and are technically and economically feasible are funded and implemented by the city government. In 2021-2022, the municipal budget allocated to such proposals has been 50 million euro.

Figures 1 and 2 show an example of a citizen proposal and its comment threads in Decide Madrid.

Figure 1: Screenshot of a Decide Madrid webpage showing a citizen proposal that suggests having more tree areas close to M-30, one of the principal motorways in Madrid.

Figure 2: Screenshot of a Decide Madrid webpage showing the comment threads of a citizen proposal.

Both proposal descriptions and comments are rich on argumentative information, which may be very valuable for citizens and government stakeholders, and which we aim to extract in our work. For this purpose, we consider the Decide Madrid dataset used by Cantador et al. (2017, 2020), which contains information about 21,744 citizen proposals —classified into 30 categories and 325 topics, geolocated in 21 city districts, and annotated with controversy scores—, and 62,838 comments.

From this dataset, we selected the 40 most controversial proposals, and collaboratively searched for and annotated the arguments given by citizens

[4]Decide Madrid, `https://decide.madrid.es`

in the proposals descriptions and comments, generating a first version of a corpus that we make publicly available[5]. To ease the manual argument annotation process, and store the identified arguments in a formal, structured format –including the components and relations of the arguments–, we used ARGAEL (Segura-Tinoco and Cantador, 2023), an easy-to-use, configurable desktop tool that we have developed to assist with the argument annotation and evaluation task, and which can be freely downloaded[6].

Our corpus is composed of 3,254 propositions annotated with 922 claims and 538 premises interconnected, and to the best of our knowledge, ours is one of the first argumentative corpora in Spanish in the AM field.

The underlying argument model is configured in ARGAEL and, going beyond the traditional *support* and *attack* argumentative links, it includes the following categories of relations between argument components (claims c and premises p):

- *Cause*, stating the *reason* or *condition* for an argument, e.g., "[The pollution levels in the city center are very high]$_c$ because [most people use the car to get around]$_p$," "[If the government wants to favor tourism]$_p$, [it must offer free tourist information]$_c$."

- *Clarification*, introducing a *conclusion*, *exemplification*, *restatement*, or *summary* of an argument, e.g., "As a conclusion, [we suggest the government to authorize this initiative]$_c$," "In short, [we have to wait for the results of the elections so that they can start to do something]$_c$."

- *Consequence*, evidencing an *explanation*, *goal*, or *result* of an argument, e.g., "[The use of public transport should be facilitated]$_p$ to [avoid pollution in the downtown area]$_c$," "[I have not seen garbage trucks for a week]$_p$, hence [the bins are full, and people have to throw the garbage in the streets]$_c$."

- *Contrast*, conflicting with an argument by giving *alternatives*, doing *comparisons*, making *concessions*, or providing *oppositions*, e.g., "On the other hand, [we must think about the costs that this work will cause due to its maintenance]$_c$," "[Restricting the access of private vehicles to the downtown area helps in mitigating noise]$_c$, but [it is still insufficient due to the presence of buses, taxis, etc.]$_c$"

- *Elaboration*, introducing an argument that provides details about another one, entailing *addition*, *precision*, or *similarity* issues, e.g., "[The asphalt of the streets is in very bad conditions]$_c$, moreover, [the sidewalks have holes]$_c$," "[The youth unemployment rate has increased compared to last year]$_c$, specifically, [it has gone from 23% to 28%]$_c$."

This taxonomy is a compendium of relations used in the AM literature (Knott and Dale, 1994; Mochales Palau and Moens, 2009b; Wei Feng and Hirst, 2011; Stab and Gurevych, 2014, 2017), and represent a fine-grained representation of argumentative structures, which entails addressing the argument relation recognition as a multi-class classification problem.

Specifically, in our corpus, we annotated 538 argument relations distributed by category as: 77 relations belonging to the *Cause* category, 64 to *Clarification*, 76 to *Consequence*, 120 to *Contrast*, and 201 to *Elaboration*. Figure 4 shows additional details about the number of argument relations by subcategory in the corpus.

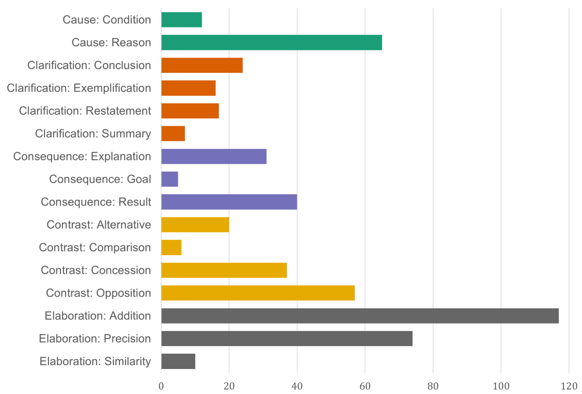

Figure 4: Number of argument relations by subcategory in our corpus.

4 Dimensionality Reduction

In this section, we introduce the linguistic features of the vector representations exploited by the evaluated ML models to AM, and the vector dimensionality reduction techniques applied before building such models.

[5]Decide Madrid corpus, https://github.com/argrecsys/decide-madrid-2019-annotations

[6]ARGAEL, https://github.com/argrecsys/argael

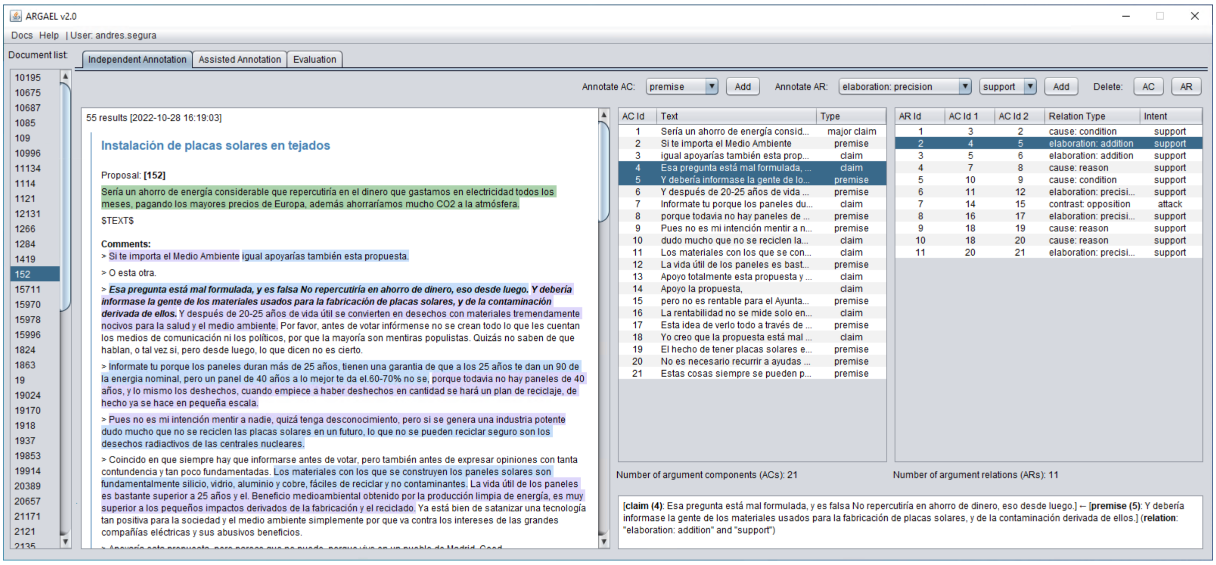

Figure 3: Screenshot of the ARGAEL tool, whose graphical user interface allows, among other things, marking argument components and relations in a set of input texts, according to a given argument model.

4.1 Linguistic Features

Researchers have considered different types of features for the AM sequence labeling tasks (Stab and Gurevych, 2014; Aker et al., 2017; Du et al., 2017; Habernal and Gurevych, 2017).

Almost all the surveyed ML-based works on argumentative fragment detection, argument component classification, and argument relation recognition make use of some well-known structural, lexical, morphosyntactic and discourse-associated features. Only Habernal and Gurevych (2017) explore the use of word embedding features (sum up the first 300 embeddings of each word, resulting in a single vector for the entire sentence) with good results in cross-domain scenarios.

Therefore, in this work we have considered the following features at sentence level:

- *Structural features*: sentence length, relative position in paragraph, average word length, number of tokens, and punctuation marks.
- *Lexical features*: bag of words 1-3 grams, TF-IDF weighted nouns, verbs and adverbs, modal auxiliaries, and named entities.
- *Morphosyntactic features*: part-of-speech 1-3 grams, depth and number of subclauses of the sentence constituency tree.
- *Discourse features*: keywords representing argumentative linkers (made publicly available with the created corpus, see Section 3).

We discard topic and sentiment features because we aimed to investigate with domain-independent features. They together with semantic and word embedding features could be explored in the future.

4.2 Dimensionality Reduction Techniques

In statistics and machine learning, dimensionality reduction is the process of reducing the number of random variables (features) under consideration, obtaining a new set of informative variables, commonly referred to as principal components.

Three of the main techniques for dimensionality reduction are Singular Value Decomposition, SVD (Beltrami, 1973; Stewart, 1993), Principal Components Analysis, PCA (Hotelling, 1933), and Linear Discriminant Analysis, LDA (Fisher, 1936).

They search for linear combinations of the features that best explain the input data, but they differ on the fact that LDA is a supervised technique that also classifies the data, and SVD and PCA are unsupervised techniques that ignore class labels.

Specifically, SVD obtains a factorization USV^t of the feature matrix A, where the diagonal entries of the small, inner matrix S are called singular values, correspond to the root of the positive eigenvalues of AA^t, and can be used as a reduced set of new variables that produce optimal low rank approximation of A with minimal reconstruction error. PCA derives new feature variables that are linear combinations of the original variables and are uncorrelated, by capturing the direction of maximum variation in the dataset, and without paying attention to the underlying class structure. Finally, LDA focuses on finding a feature subspace that maximizes the separability of classes, i.e., finding directions (components) of maximum variance.

As shown by Martínez and Kak (2001), although it is generally believed that algorithms based on LDA are superior to those based on PCA, this is not always the case. On the image recognition field, the authors concluded that if the target dataset is small, PCA can outperform LDA, being less sensitive to the used training set. In fact, as we shall show in the experimental section, LDA degraded the addressed AM tasks since its resultant components are determined by the number of classes to predict: 2 for the argumentative fragment detection task, 3 for the argument component classification task, and 6 for the argument relation recognition task.

5 Experiments

In this section, we describe the methodology used to evaluate a number of machine learning models for the three target AM tasks (i.e., *argumentative fragment detection*, *argument component classification* and *argument relation recognition*), and report their performance results with and without dimensionality reduction of the input data.

5.1 Evaluation Methodology

We approached the *argumentative fragment detection* and *argument component classification* tasks as binary classification problems –*argumentative* vs. *non-argumentative*), and *premise* vs. *claim*, respectively–, and the *argument relation recognition* as a multi-class classification problem, with a total of six relation types (classes): *cause*, *consequence*, *contrast*, *elaboration*, *clarification*, and *none* (in absence of relation).

The *argument component classification* task (task 2) was tested on those feature vectors that corresponded to text fragments previously identified as argumentative (task 1).

The *argument relation recognition* task (task 3) was tested on vectors obtained by concatenating each pair of argumentative text fragment vectors (from task 1), considering their order. That is, given two argument components c_1 and c_2, task 3 was fed with two vectors $u = (c_1, c_2)$ and $v = (c_2, c_1)$. If the argument components were linked via a relation r, e.g., (c_2, r, c_1), one of such vectors (v in the example) was assigned with a class that correspond to the type of r, whereas the other vector was assigned with the *none* class.

Considering the surveyed related work of Section 2, the ML algorithms we selected for the above tasks are: naive Bayes (NB), logistic regression (LR), support vector machine (SVM), and gradient boosting (GB).

For all tasks, we split the dataset into 80% for training and 20% for testing. We followed a stratified data split with respect to the label to be predicted, and used the 10-fold cross-validation technique on the training data to find the best hyperparameters for the ML algorithms. Before splitting, we normalized the input values of each feature to the [0,1] range.

The optimal values of the hyperparameters of the classification models and the SVD/PCA/LDA techniques were obtained by grid search with respect to the micro-F1 score. As future work, we leave the use of other more efficient model training optimization methods, such as Optuna, presented by Akiba et al. (2019). Table 1 shows the hyperparameters configurations tested, and their optimal values obtained for each ML algorithm and AM task.

In each ML algorithm training configuration, we tested several numbers of principal components for the dimensionality reduction techniques, in order to explore whether horizontal reduction of the input feature space improved classification performance. Specifically, for SVD and PCA, we tested 20 different numbers of components, from 25 up to 500 (with increment steps of 25). In the case of LDA, for each AM task, the number of dimensions was reduced to the number of target classes minus 1.

For the tested number of principal components, figures 5, 6 and 7 show the effects of the SVD and PCA dimensionality reduction techniques on the performance (in terms of micro-F1 score values) of the tested ML algorithms in the three AM tasks. As it can be seen, in general, the ML algorithms outperformed their counterparts operating on reduced feature spaces and, as expected, the F1 tends to increase with the number of components. We discuss the maximum performance values for all approaches in the next subsection. More details are given in Appendix A.

5.2 Classification Results

Tables 2, 3 and 4 show the best performance results of the evaluated approaches, for *argumen-*

Algorithm	Hyperparameter	Tested values	Task 1	Task 2	Task 3
NB	alpha	{0.0001, 0.001, 0.01, 0.1, 1}	1	1	1
	fit prior	{true, false}	false	false	false
LR	solver	{newton-cg, lbfgs, liblinear, saga}	liblinear	saga	saga
	C	{0.001, 0.01, 0.1, 1, 10, 100}	0.1	1	1
	penalty	{none, elasticnet, L1, L2}	L2	L2	L1
SVM	kernel	{linear, rbf}	rbf	linear	rbf
	C	{100, 10, 1, 0.1, 0.01, 0.001}	10	0.1	10
	gamma	{1, 0.1, 0.01, 0.001, 0.0001}	0.01	-	0.1
GB	learning rate	{0.15, 0.1, 0.05, 0.02, 0.01}	0.1	0.1	0.1
	n estimators	{150, 175, 200, 225, 250}	200	200	150
	max depth	{2, 3, 4, 5, 6}	3	3	5
	min samples leaf	{1, 2, 5, 7}	5	5	1
	min samples split	{2, 3}	2	2	2

Table 1: Tested hyperparameter values and obtained best performing hyperparameters for each ML algorithm and AM task: argument detection (task 1), argument component classification (task 2), and argument relation recognition (task 3). The names of the hyperparameters are those used by the Python Scikit-learn library. NB, LR, SVM and GB stand for Naive Bayes, Logistic Regression, Support Vector Machine, and Gradient Boosting, respectively.

tative fragment detection, *argument component classification* and *argument relation recognition*, respectively. They report the accuracy (acc), precision (p), recall (r) and micro-F1 score (F1) values of the ML algorithms on the original feature spaces and on the principal component spaces obtained by SVD and PCA. The results of LDA are not reported because they were relatively low. This supervised technique degraded the resultant component space, whose dimension was determined by the number of classes in the target AM tasks.

The results show that reducing the dimensionality of the corpus feature space –composed of a total of 12,593 lexical, morphosyntactic, structural and discourse-associated features extracted from each sentence–, did not impact drastically on the classification accuracy of the evaluated ML models. Applying dimensionality reduction, the best reached F1 was similar to the best F1 achieved by using the entire feature space: on average, 97% for task 1, 95% for task 2, and 107% for task 3. In some cases (which are underlined in the tables), the F1 values achieved by the ML algorithm on a reduced space were greater than the entire space ones.

In particular, we observe that the first 400-500 components of SVD and PCA provided the best relative performance on the *Logistic Regression* and *Support Vector Machine* algorithms. This represents around 3-4% of the number of dimensions in the entire input feature space. Thus, in terms of training time, we found remarkable improvements for the three tasks, reducing on average the time required to train the ML algorithms by 77.58%, 84.29% and 82.31%, respectively for tasks 1, 2 and 3. This finding, although expected, is significant, as it would allow testing a larger hyperparameter set and fast experimenting with new algorithmic solutions, while reducing the well-known carbon footprint generated by massive ML model training.

As shown in the tables, when no dimensionality reduction was applied, GB was consistently the best performing algorithm, achieving decreasing maximum F1 values for the three consecutive tasks: 0.729, 0.624 and 0.554 (marked in bold in the tables). These values reflect the increasing difficulty of the underlying classification problems.

In the *argumentative fragment detection* (binary classification) task, GB achieved the best performance (F1=0.729), closely followed by NB (F1=0.726). SVM was the worst performing ML algorithm. However, this algorithm took benefit from the dimensionality reduction techniques, especially from SVD, with which it was able to increase its F1 value to 0.725, using its first n=500 components (4% of the total number of original dimensions).

For the *argument component classification* (multiclass) task, GB and SVM again were respectively the best and worst performing ML algorithms, with maximum F1 values equal to 0.624 and 0.584. In this case, LR was the algorithm whose performance improved the most with the help of the dimensionality reduction techniques; specifically, it reached an F1 value of 0.620 with the first n=400 principal components of PCA, representing 3% of the number of original features.

Finally, with respect to the *argument relation recognition* (multiclass) task, GB –with an F1 value of 0.554– was followed in performance by approaches that made use of dimensionality reduction

Approach	acc	p	r	F1
NB	.727	.726	.727	.726
NB + SVD (n=250)	.647	.656	.647	.647
NB + PCA (n=425)	.633	.649	.633	.632
LR	.717	.717	.717	.715
LR + SVD (n=400)	.708	.708	.708	.705
LR + PCA (n=350)	.708	.708	.708	.705
SVM	.711	.711	.711	.708
SVM + SVD (n=500)	.727	.726	.727	.725
SVM + PCA (n=475)	.719	.718	.719	.717
GB	.730	.729	.730	**.729**
GB + SVD (n=325)	.719	.721	.719	.714
GB + PCA (n=350)	.710	.711	.710	.705

Table 2: Achieved results on the argumentative fragment detection task.

Approach	acc	p	r	F1
NB	.624	.589	.624	.587
NB + SVD (n=200)	.499	.573	.499	.521
NB + PCA (n=150)	.518	.594	.518	.539
LR	.633	.603	.633	.607
LR + SVD (n=500)	.636	.612	.636	.615
LR + PCA (n=400)	.642	.618	.642	.620
SVM	.621	.586	.621	.584
SVM + SVD (n=400)	.624	.608	.624	.570
SVM + PCA (n=425)	.627	.612	.627	.573
GB	.648	.631	.648	**.624**
GB + SVD (n=100)	.604	.577	.604	.556
GB + PCA (n=125)	.594	.556	.594	.551

Table 3: Achieved results on the argument component classification task.

Approach	acc	p	r	F1
NB	.490	.363	.490	.355
NB + SVD (n=500)	.455	.472	.455	.462
NB + PCA (n=475)	.470	.488	.470	.477
LR	.555	.537	.555	.489
LR + SVD (n=500)	.555	.483	.555	.490
LR + PCA (n=300)	.545	.456	.545	.470
SVM	.570	.643	.570	.472
SVM + SVD (n=225)	.525	.474	.525	.482
SVM + PCA (n=275)	.535	.488	.535	.490
GB	.615	.594	.615	**.554**
GB + SVD (n=350)	.550	.510	.550	.482
GB + PCA (n=125)	.555	.552	.555	.490

Table 4: Achieved results on the argument relation classification task.

for all the reminder ML algorithms. Thus, this task, despite being the most complex of the three main AM tasks, was the one that took the most advantage from using the unsupervised SVD and PCA techniques.

6 Conclusions

Although the conducted experiments can be considered preliminary, they have shown promising results about the potential benefits of selecting informative linguistic features and reducing dimensionality in ML-based approaches to AM. For the three major AM tasks (i.e., argumentative fragment detection, argument component classification, and argument relation recognition), and for almost all the ML algorithms used in the AM literature, working on feature spaces of much lower dimensionality generated by SVD and PCA has entailed not only improvements in training efficiency, but also consistent classification performance of the algorithms, especially logistic regression and support vector machines.

In addition to these issues, our work contributes to the AM field through the publication of a new argumentative corpus in Spanish on e-participation, a novel and relevant domain for the AM community. We plan to increase the size and quality of the corpus, and hope it will be of interest for researchers and practitioners. Regardless of the impact of dimensionality reduction, the developed AM methods and their performance results on our corpus could be of reference for future improvements.

Moreover, we believe that the corpus may be exploited in different argumentative scenarios. In particular, it could be used to extract argumentative threads from online political discussions (Lawrence et al., 2017) and parliamentary debates, whose transcripts are available as open government datasets.

7 Limitations

As previous work on machine learning-based AM, a limitation of our study is the fact that we have aimed to extract argumentative units at sentence level. However, a single sentence may contain several units, such as a claim and an associated premise, and an argumentative unit could encompass several, generally two, consecutive sentences (Habernal and Gurevych, 2017).

Moreover, to draw robust and generalizable conclusions about the advantages of applying dimensionality reduction, we need to make further experiments not only with more data, but also considering other types of features (e.g., word embeddings) and several domains and corpora, which may be in languages distinct to Spanish (Lawrence and Reed, 2020).

We could further research which are the most relevant features in each of the AM tasks, and focus on and boost them with ad hoc algorithms. In this context, we could also consider topic, sentiment, debate structure, and domain (or language) dependent features that may be valuable to identify argumentative fragments and their components and relations (Lawrence and Reed, 2020).

Finally, we should compare our results with those achieved by of recent deep learning approaches to argument extraction (Eger et al., 2017; Reimers et al., 2019), in order to properly analyze the benefits and drawbacks of using a simple technique with respect to much more complex and computational costly methods. For such purpose, we have to extend our corpus, so that deep learning architectures for AM could be fine-tuned with existing large language models, such as BETO (Cañete et al., 2020) for the Spanish language.

Acknowledgements

This research was supported by the Spanish Ministry of Science and Innovation: Grant PID2019-108965GB-I00 and Grant PID2022-139131NB-I00, funded by MCIN/AEI/10.13039/501100011033 and by "ERDF A way of making Europe." We sincerely thank the anonymous reviewers of the paper for their valuable comments and suggestions.

References

Ahmet Aker, Alfred Sliwa, Yuan Ma, Ruishen Lui, Niravkumar Borad, Seyedeh Ziyaei, and Mina Ghobadi. 2017. What works and what does not: Classifier and feature analysis for argument mining. In *Proceedings of the 4th Workshop on Argument Mining*, pages 91–96.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, page 2623–2631. Association for Computing Machinery.

Alaa Alhamzeh, Mohamed Bouhaouel, Elöd Egyed-Zsigmond, Jelena Mitrović, Lionel Brunie, and Harald Kosch. 2021. A stacking approach for cross-domain argument identification. In *Proceedings of the 32nd International Conference on Database and Expert Systems Applications*, pages 361–373. Springer.

Eugenio Beltrami. 1973. Sulle funzioni bilineari. *Giornale di matematiche ad Uso degli Studenti Delle Università*, 11:98–106. An English translation by D. Boley is available at the Department of Computer Science of University of Minnesota, Minneapolis, MN, Technical Report 90-37, 1990.

Abdelhamid Boudjelida, Sehl Mellouli, and Jungwoo Lee. 2016. Electronic citizens participation: Systematic review. In *Proceedings of the 9th International Conference on Theory and Practice of Electronic Governance*, pages 31–39.

Syed Burhan ud Din Tahir. 2017. Comparative analysis of supervised learning approaches for argument identification. In *Proceedings of the 20th International Multi-topic Conference*, pages 1–5. IEEE.

José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. 2020. Spanish pre-trained BERT model and evaluation data. In *Proceedings of the 2020 Workshop on Practical ML for Developing Countries*, volume 2020, pages 1–10.

Iván Cantador, Alejandro Bellogín, María E Cortés-Cediel, and Olga Gil. 2017. Personalized recommendations in e-participation: Offline experiments for the 'decide madrid' platform. In *Proceedings of the International Workshop on Recommender Systems for Citizens*, pages 1–6. CEUR Workshop Proceedings.

Iván Cantador, María E Cortés-Cediel, and Miriam Fernández. 2020. Exploiting open data to analyze discussion and controversy in online citizen participation. *Information Processing & Management*, 57(5):102301.

Mamoru Deguchi and Kazunori Yamaguchi. 2019. Argument component classification by relation identification by neural network and TextRank. In *Proceedings of the 6th Workshop on Argument Mining*, pages 83–91. ACL.

Yang Du, Minglan Li, and Mengxue Li. 2017. Joint extraction of argument components and relations. In *Proceedings of the 2017 International Conference on Asian Language Processing*, pages 1–4. IEEE.

Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. 2017. Neural end-to-end learning for computational argumentation mining. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, page 11–22. ACL.

Ronald A Fisher. 1936. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188.

Marc Gramberger. 2001. *Citizens as partners. OECD Handbook on Information, Consultation and Public Participation in Policy-making*. OECD Publishing.

Ivan Habernal and Iryna Gurevych. 2017. Argumentation mining in user-generated web discourse. *Computational Linguistics*, 43(1):125–179.

David Held. 2006. *Models of Democracy*. Polity.

Harold Hotelling. 1933. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417.

Alistair Knott and Robert Dale. 1994. Using linguistic phenomena to motivate a set of coherence relations. *Discourse Processes*, 18(1):35–62.

Anang Kunaefi and Masayoshi Aritsugi. 2020. Characterizing user decision based on argumentative reviews. In *Proceedings of the 2020 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies*, pages 161–170. IEEE.

John Lawrence, Joonsuk Park, Katarzyna Budzynska, Claire Cardie, Barbara Konat, and Chris Reed. 2017. Using argumentative structure to interpret debates in online deliberative democracy and eRulemaking. *ACM Transactions on Internet Technology*, 17(3):1–22.

John Lawrence and Chris Reed. 2020. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Marco Lippi and Paolo Torroni. 2015. Argument mining: A machine learning perspective. In *Proceedings of the 3rd International Workshop on Theory and Applications of Formal Argumentation*, pages 163–176. Springer.

Marco Lippi and Paolo Torroni. 2016. Argumentation mining: State of the art and emerging trends. *ACM Transactions on Internet Technology*, 16(2):1–25.

Anastasios Lytos, Thomas Lagkas, Panagiotis Sarigiannidis, and Kalina Bontcheva. 2019. The evolution of argumentation mining: From models to social media and emerging tools. *Information Processing & Management*, 56(6):102055.

Aleix M Martínez and Avinash C Kak. 2001. PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):228–233.

Tobias Mayer, Elena Cabrio, and Serena Villata. 2020. Transformer-based argument mining for healthcare applications. In *Proceedings of the 24th European Conference on Artificial Intelligence*, pages 2108–2115. IOS Press.

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

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

Raquel Mochales Palau and Marie-Francine Moens. 2011. Argumentation mining. *Artificial Intelligence and Law*, 19(1):1–22.

Prakash Poudyal, Teresa Goncalves, and Paulo Quaresma. 2016. Experiments on identification of argumentative sentences. In *Proceedings of the 10th International Conference on Software, Knowledge, Information Management & Applications*, pages 398–403. IEEE.

Nils Reimers, Benjamin Schiller, Tilman Beck, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. 2019. Classification and clustering of arguments with contextualized word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 567–578. ACL.

Andrés Segura-Tinoco and Iván Cantador. 2023. ARGAEL: ARGument Annotation and Evaluation tooL. *SoftwareX*, 23:101410.

Christian Stab and Iryna Gurevych. 2014. Identifying argumentative discourse structures in persuasive essays. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 46–56.

Christian Stab and Iryna Gurevych. 2017. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659.

Gilbert W Stewart. 1993. On the early history of the singular value decomposition. *SIAM review*, 35(4):551–566.

Vanessa Wei Feng and Graeme Hirst. 2011. Classifying arguments by scheme. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 987–996. ACL.

A Effects of the Dimensionality Reduction

Figures 5, 6 and 7 show the effects of the 3 dimensionality reduction techniques used on the performance (in terms of micro-F1 score values) of the tested ML algorithms, in the *argumentative fragment detection*, *argument component classification* and *argument relation recognition* tasks, respectively.

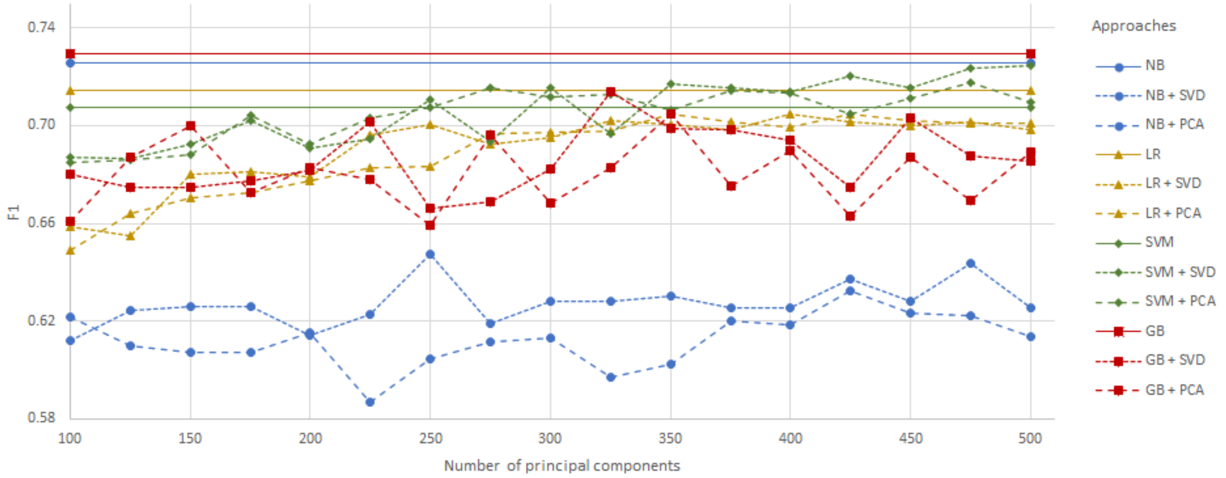

Figure 5: Micro-F1 score values achieved on the *argumentative fragment detection* task on training sets. NB, LR, SVM and GB stand for Naive Bayes, Logistic Regression, Support Vector Machines, and Gradient Boosting, respectively.

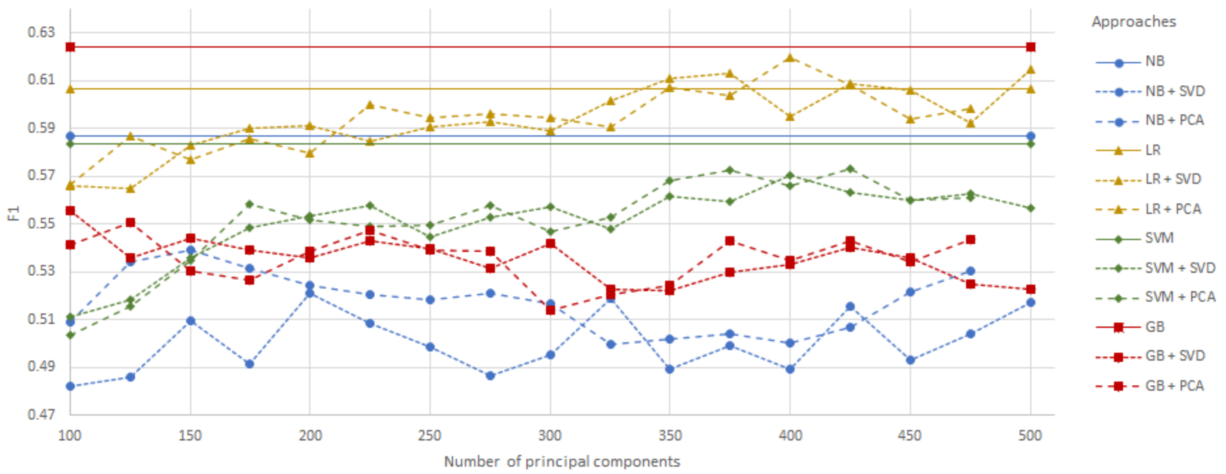

Figure 6: Micro-F1 score values achieved by the tested approaches on the *argument component classification* task on training sets. NB, LR, SVM and GB stand for Naive Bayes, Logistic Regression, Support Vector Machines, and Gradient Boosting, respectively.

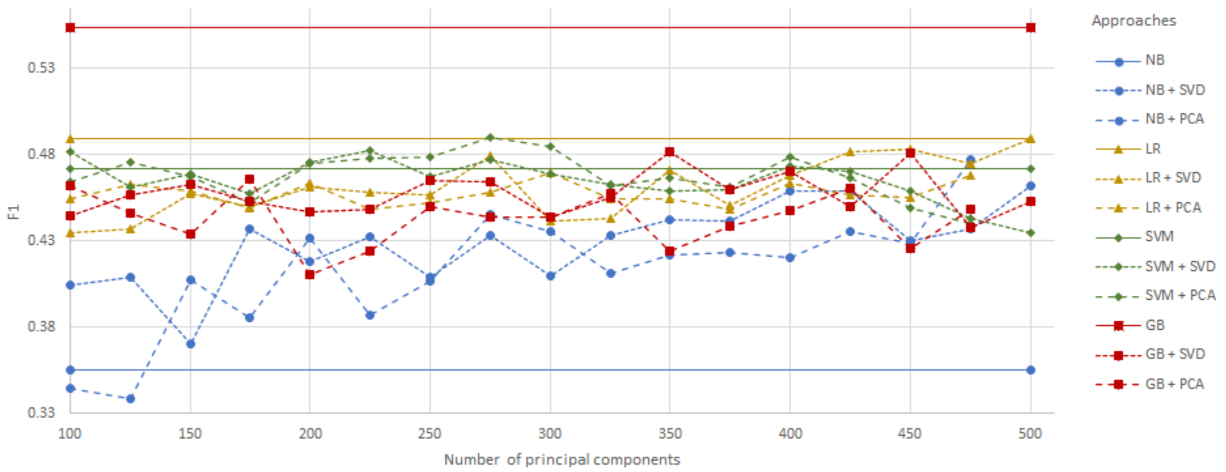

Figure 7: Micro-F1 score values achieved by the tested approaches on the *argument relation recognition* task on training sets. NB, LR, SVM and GB stand for Naive Bayes, Logistic Regression, Support Vector Machines, and Gradient Boosting, respectively.