Procesamiento del Lenguaje Natural, Revista nº 47 septiembre de 2011, pp 207-214 recibido 03-05-2011 aceptado 27-05-2011

Performance analysis of Particle Swarm Optimization applied to unsupervised categorization of short texts

Análisis de Prestación de Particle Swarm Optimization aplicado a Categorización no Supervisada de Textos Cortos

Leticia Cagnina, Diego Ingaramo,
Marcelo Errecalde
LIDIC Research Group
Universidad Nacional de San Luis
Ej. de los Andes 950
5700 San Luis, Argentina.
{lcagnina,daingara,merreca}@unsl.edu.ar

Paolo Rosso
Natural Language Engineering Lab.
ELiRF, DSIC
Universidad Politécnica de Valencia
Camino de Vera s/n
46022 Valencia, España.
prosso@dsic.upv.es

Resumen: Existe actualmente la necesidad de acceder a información en línea tal como resúmenes, noticias, opiniones, evaluaciones de productos, etc. Dicha información está disponible en la web, generalmente con el formato de textos cortos. Trabajos previos han demostrado la efectividad de un algoritmo discreto Particle Swarm Optimization, llamado CLUDIPSO, para el agrupamiento de colecciones pequeñas de textos cortos. Este artículo presenta un estudio preliminar sobre la prestación de CLUDIPSO con colecciones más grandes. Los resultados fueron comparados con los obtenidos con algoritmos representativos del estado del arte en el área. El trabajo experimental muestra una fuerte evidencia sobre los inconvenientes que posee el algoritmo cuando debe agrupar colecciones de mayor tamaño. Con respecto a este último aspecto, se discuten posibles razones del comportamiento inadecuado de CLUDIPSO y se consideran algunas alternativas para resolver los problemas observados.

Palabras clave: Categorización no supervisada, Textos Cortos, Optimización mediante Cúmulo de Partículas

Abstract: Nowadays there is a need to access to on line information such as abstracts, news, opinions, evaluations of products, etc. That information is generally available on the web as short texts. Previous works have demonstrated the effectiveness of a discrete Particle Swarm Optimization algorithm, named CLUDIPSO, for clustering small short-text corpora. This article presents a preliminary study about the performance of CLUDIPSO on larger short-text corpora. The results were compared with those of the most representative algorithms of the state-of-the-art in the area. The experimental work gives strong evidence about the drawbacks of this algorithm to manage larger corpora. With respect to this last aspect, some possible reasons about the poor behavior of CLUDIPSO with larger short texts corpora are discussed and some alternatives in order to solve the problems observed, are considered.

Keywords: Unsupervised Categorization, Short Texts, Particle Swarm Optimization

1 Introduction

Web repositories are full of documents that people usually need to access. Those documents frequently are short texts, with just tens or hundreds of words, that require to be grouped for purposes to be submitted to web users. This need has caused that document clustering becomes a fundamental process in many task related to information retrieval on the web.

The main goal in a document clustering problem is to assign a set of documents into different clusters. In this context, the clustering of short-text corpora, is one of the most difficult tasks in natural language processing due to the low frequencies of terms in the documents.

In document clustering, the information about categories and correctly categorized documents is not provided in advance. An

ISSN 1135-5948

important consequence of this lack of information is that in realistic document clustering problems, the results can not usually be evaluated with typical *external* measures like F-Measure and the Entropy, because the correct categorizations specified by a human expert are not available. Therefore, the quality of the resulting groups is evaluated with respect to *structural* properties expressed in different *Internal Clustering Validity Measures* (ICVMs). Classical ICVMs used as cluster validity measures include the Dunn and Davies-Bouldin indexes, the *Global Silhouette* (GS) coefficient and, new graph-based measures such as the *Expected Density Measure* (EDM) and the λ-Measure (see (Ingaramo et al., 2008) for detailed descriptions of these ICVMs).

The use of these unsupervised measures of cluster validity -or any arbitrary criterion function that gives a reasonable estimation of the quality of the obtained groups- is not limited to the cluster evaluation stage. They can also be used as an *objective function* that the clustering algorithm attempts to optimize *during* the grouping process. This approach has been adopted by clustering algorithms like CLUDIPSO, a discrete Particle Swarm Optimizer (PSO) which obtained in previous works (Ingaramo et al., 2009) interesting results on small short-text corpora, using the GS coefficient as objective function.

This article reports an experimental study related to the performance of CLUDIPSO on short-text corpora of different sizes. The aim of this investigation is to detect possible limitations of this algorithm to scale up to larger corpora than those considered in the initial studies. In order to analyze this aspect, CLUDIPSO was compared with some of the most effective clustering algorithms in the area and with a representative number of corpora of different sizes. The experimental work confirmed the good performance of CLUDIPSO on small corpora, but it also showed some limitations to deal with larger corpora. The present work poses some possible reasons of the poor behavior of CLUDIPSO in these cases and also describes some works that are currently being developed in order to improve the CLUDIPSO performance on larger short-text corpora.

The remainder of the paper is organized as follows. Section 2 describes the PSO-based algorithm under study: CLUDIPSO. Section 3 describes some general features of the corpora used in the experiments. The experimental setup and the analysis of the results obtained from the empirical study is provided in Section 4. Finally, some general conclusions are drawn and present and future work is discussed in Section 5.

2 The CLUDIPSO Algorithm

2.1 The basic PSO algorithm

CLUDIPSO (CLUstering with a DIscrete Particle Swarm Optimization), is based on a PSO (Eberhart and Kennedy, 1995) algorithm that operates on a population of particles. Each particle is a real numbers vector which represents a position in the search space defined by the variables corresponding to the problem to solve. The best position found so far for the swarm (*gbest*) and the best position reached by each particle (*pbest*) are recorded at each cycle (iteration of the algorithm). The particles evolve at each cycle using two updating formulas, one for velocity (Equation (1)) and another for position (Equation (2)).

$$v_{id} = w(v_{id} + \gamma_1(pb_{id} - par_{id}) + \gamma_2(pg_d - par_{id})) \quad (1)$$

$$par_{id} = par_{id} + v_{id} \quad (2)$$

where par_{id} is the value of the particle i at the dimension d, v_{id} is the velocity of particle i at the dimension d, w is the inertia factor (Shi and Eberhart, 1998) whose goal is to balance global exploration and local exploitation, γ_1 is the personal learning factor, and γ_2 the social learning factor, both multiplied by 2 different random numbers within the range [0, 1]. pb_{id} is the best position reached by the particle i and pg_d is the best position reached by any particle in the swarm.

2.2 A PSO discrete version

In CLUDIPSO, each valid clustering is represented with a particle. The particles are n-dimensional integer vectors, where n is the number of documents in the corpus. Figure 1 illustrates a valid clustering (represented by the particle) of n documents which were grouped in 3 different clusters. Since the task was modeled with a discrete approach, a new formula was developed for updating the positions (shown in Equation (3)).

$$par_{id} = pb_{id} \quad (3)$$

where par_{id} is the value of the particle i at the dimension d and pb_{id} is the best position reached by the particle i until that moment. This equation was introduced with the objective of accelerate the convergence velocity of the algorithm (principal incoming of discrete PSO models). It is important to note that in this approach the process of updating particles is not as direct as in the continuous case (basic PSO algorithm). In CLUDIPSO, the updating process is not carried out on all dimensions at each iteration. In order to determine which dimensions of a particle will be updated the following steps are performed:

1. all dimensions of the velocity vector are normalized in the [0, 1] range, according to the process proposed by Hu et al. (Hu, Eberhart, and Shi, 2003) for a discrete PSO version;
2. a random number $r \in [0, 1]$ is calculated;
3. all the dimensions (in the velocity vector) higher than r are selected in the position vector, and updated using the Equation (3).

To help avoiding convergence to a local optimum, a dynamic mutation operator (Cagnina, Esquivel, and Gallard, 2004) is used, which is applied to each individual with a pm-probability. This value is calculated considering the total number of iterations in the algorithm (*cycles*) and the current cycle number as the Equation (4) indicates:

$$pm = max\_pm - \frac{max\_pm - min\_pm}{max\_cycle} * current\_cycle \quad (4)$$

where max_pm and min_pm are the maximum and minimum values that pm can take, max_cycle is the total number of cycles that the algorithm will iterate, and $current\_cycle$ is the current cycle in the iterative process. The mutation operation is applied if the particle is the same that its own *pbest*, as was suggest by (Hu, Eberhart, and Shi, 2003). The mutation operator swaps two random dimensions of the particle.

2.3 The CLUDIPSO objective function

The objective function to be optimized with CLUDIPSO is GS because it has shown an adequate correlation degree with the categorization of a human expert. GS measure combines two key aspects to determine the quality of a given clustering: *cohesion* and *separation*. Cohesion measures how closely related are objects in a cluster whereas separation quantifies how distinct (well-separated) a cluster from another is. The GS coefficient of a clustering is the average cluster silhouette of all the obtained groups. The cluster silhouette of a cluster C also is an average silhouette coefficient but, in this case, of all objects belonging to C. Therefore, the fundamental component of this measure is the formula used for determining the silhouette coefficient of any arbitrary object i, refered as $s(i)$ and is defined as follows:

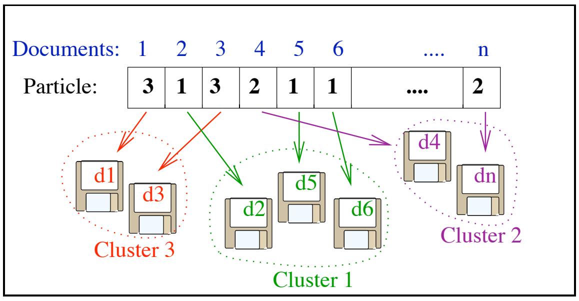


Figure 1: CLUDIPSO particle representing a clustering of n documents in 3 clusters.

$$s(i) = \frac{(b(i) - a(i))}{max(a(i), b(i))}$$

with $-1 \leq s(i) \leq 1$. The $a(i)$ value denotes the average dissimilarity of the object i to the remaining objects in its own cluster, and $b(i)$ is the average dissimilarity of the object i to all objects in the nearest cluster. From this formula it can be observed that negative values for this measure are undesirable and values close to 1 are the best.

3 Data Sets

The inherent difficulty of short-document clustering problems requires a detailed analysis of the features of each corpus used in the experiments. For this reason, some specific characteristics such as number of documents, terms and groups of the used corpora are considered below.

In this experimental work the *Micro4News*, *EasyAbstracts*, *SEPLN-CICLing* and *CICLing-2002* short-text corpora were selected. The documents include topics of news and abstracts of scientific papers. The corpora are considered small because they only have 48 documents. Several works (Makagonov,

Corpora	**|Doc|**	**|T|**	**|G|**
Micro4News	48	125614	4
EasyAbstracts	48	9261	4
SEPLN-CICLing	48	3143	4
CICLing-2002	48	3382	4
R4	266	27623	4
R6	536	53494	6
R8B	816	71842	8
JRC-Acquis	563	1424074	6

Table 1: Main characteristics of selected corpora.

Alexandrov, and Gelbukh, 2004; Alexandrov, Gelbukh, and Rosso, 2005; Pinto, Benedí, and Rosso, 2007; Ingaramo et al., 2008) have used these corpora to test the performance of their approaches and the interested reader can obtain more information about them in (Errecalde and Ingaramo, 2008).

Four larger corpora (with different characteristics) were also considered: *R4*, *R6*, *R8B* and *JRC-Acquis*. The first three are subsets of the well known *R8-Test* corpus, a subcollection of the *Reuters-21578* (Frank and Asuncion, 2010) dataset with news. *JRC-Acquis* is a subcollection of Acquis (Steinberger et al., 2006), a popular corpus with legislative documents of European Union.

The main differences between the all corpora are shown in table 1 in which information about number of documents (|D|), terms (|T|) and groups (|G|) of each corpus can be obtained.

4 Experimental Study

4.1 Results obtained

In the experiments, 50 independent experiments per corpus were performed, with 10,000 iterations (*cycles*) per run. CLUDIPSO used the following parameters: swarm size = 50 particles, dimensions at each particle = number of documents (|Doc|), *pm_min* = 0.4, *pm_max* = 0.9, inertia factor $w = 0.9$, personal and social learning factors for γ_1 and γ_2 were set to 1.0. The parameter settings such as swarm size, mutation probability and learning factors were empirically derived after several experiments. It is important to note that for larger corpora, the algorithm was tested with more iterations and more particles but the improvements were not substantial compared to the increase in the execution time of a single experiment. The results were compared with those obtained with other three clustering algorithms: K-Means (MacQueen, 1967), K-MajorClust (Stein and Niggemann, 1999) and CHAMELEON (Karypis, Han, and Kumar, 1999). K-Means is one of the most popular clustering algorithms and, K-MajorClust and CHAMELEON are representative of the density-based and graph-based approaches to the clustering problem. Information about the correct number of groups (k) has to be provided to the algorithms.

Algorithms	F_{min}	F_{max}	F_{min}	F_{max}
	Micro4News		*EasyAbstracts*	
K-Means	0.41	0.96	0.31	0.71
K-MajorClust	**0.94**	0.96	0.48	**0.98**
CHAMELEON	0.46	0.96	0.39	0.96
CLUDIPSO	0.85	**1**	**0.85**	**0.98**
	SEPLN-CICLing		*CICling-2002*	
K-Means	0.36	0.69	0.35	0.6
K-MajorClust	0.52	0.75	0.36	0.48
CHAMELEON	0.4	0.76	0.38	0.52
CLUDIPSO	**0.58**	**0.85**	**0.47**	**0.73**

Table 2: F-Measure values for small corpora.

The quality of the results was evaluated using the classical (external) F-measure on the clusterings that each algorithm generated in 50 independent experiments per corpus. The reported results correspond to the minimum (F_{min}) and maximum (F_{max}) F-measure values. Tables 2 and 3 show the F_{min} and F_{max} values obtained with the selected corpora. The values highlighted in bold indicate the best minimum and maximum values obtained with the considered corpora.

With the small corpora (less than 50 documents) it is observed in Table 2 that CLUDIPSO obtained the best F_{max} values and, in some cases, with a notable difference with respect to the other tested algorithms (see for instance, the results with *SEPLN-CICLing* and *CICLing-2002*). Similar results can be observed with the F_{min} values in corpora like *EasyAbstracts*, *SEPLN-CICLing* and *CICLing-2002* in which the minimum values of CLUDIPSO clearly outperformed those of the remaining algorithms. It is worth noting that the highest possible value of F_{max} (which is 1 and means the perfect classification) was reached by CLUDIPSO with *Micro4News* although the best F_{min} value for

Algorithms	F_{min}	F_{max}	F_{min}	F_{max}
	R4		*R6*	
K-Means	**0.57**	**0.91**	**0.51**	**0.81**
K-MajorClust	0.45	0.79	0.36	0.74
CHAMELEON	0.47	0.83	0.42	0.66
CLUDIPSO	0.48	0.75	0.26	0.38
	R8B		*JRC-Acquis*	
K-Means	0.48	**0.78**	**0.40**	**0.64**
K-MajorClust	0.28	0.68	0.33	0.55
CHAMELEON	**0.57**	0.71	0.31	0.56
CLUDIPSO	0.18	0.25	0.26	0.33

Table 3: F-Measure values for larger corpora.

this corpora was obtained by K-MajorClust. These results are conclusive with respect to the good performance that CLUDIPSO can obtain with small short-text corpora.

For larger short-text corpora (Table 3) such as Reuters-derived and *JRC-Acquis*, CLUDIPSO obtained poor results in all cases, being K-Means the algorithm that generally achieved the best results. Additional information about the poor behavior of CLUDIPSO with the larger corpora can be obtained from the Boxplots (Tukey, 1977) with the distribution of F-measure values (averaged) shown in Figure 2 [1]. Boxplots display graphically differences between samples (results of the experiments) using a box (the size indicates the data dispersion), divided into 25th and 75th percentiles with a line (the median value). Vertical lines outside the box indicate smallest and largest observations (the whiskers) and outlier values are marked with dots.

In Figure 2, the results obtained by CLUDIPSO and K-Means with *R4* showed some dispersion. This means that both algorithms did not obtain similar results in the total of executions done. The median value in the boxplot of CLUDIPSO presents a strong bias to the right side showing that few of the best values in all executions are around 0.65. The median value of K-Means is slightly better than that obtained with CLUDIPSO (around 0.7) and K-MajorClust does not evidence a big dispersion but all values in all executions are lower than those of CLUDIPSO and K-Means. The boxplot of K-Means has similar whisker as CLUDIPSO ones. Then, studying the distribution of averaged F-Measure values, the boxplots do not show a big difference of performance between CLUDIPSO and K-Means although the last algorithm obviously outperforms the first one.

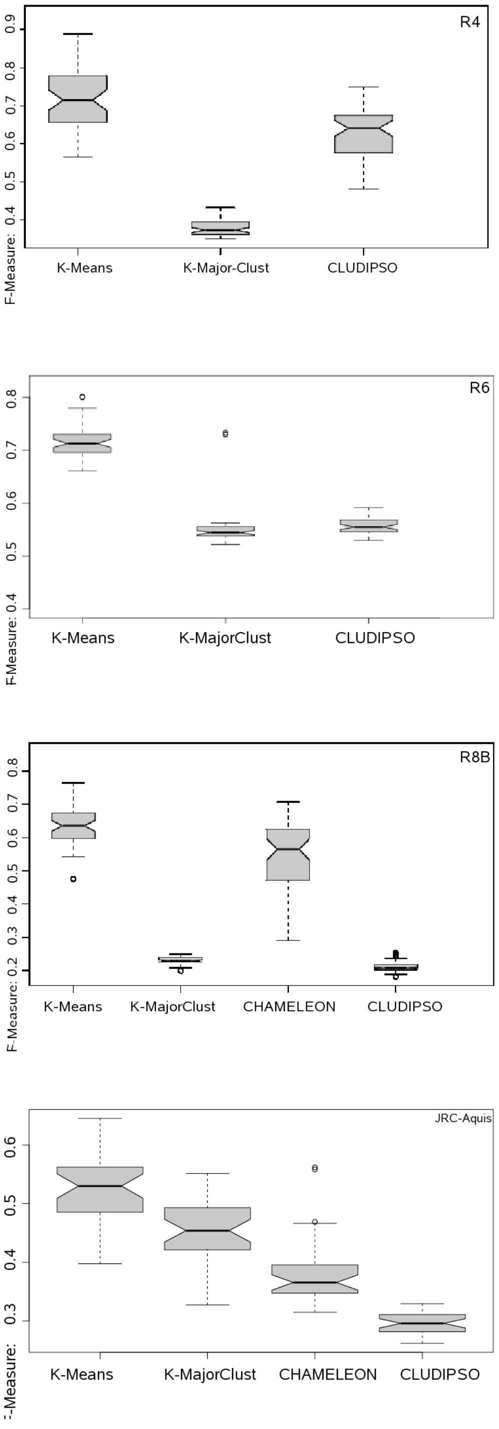


Figure 2: Boxplots for larger corpora.

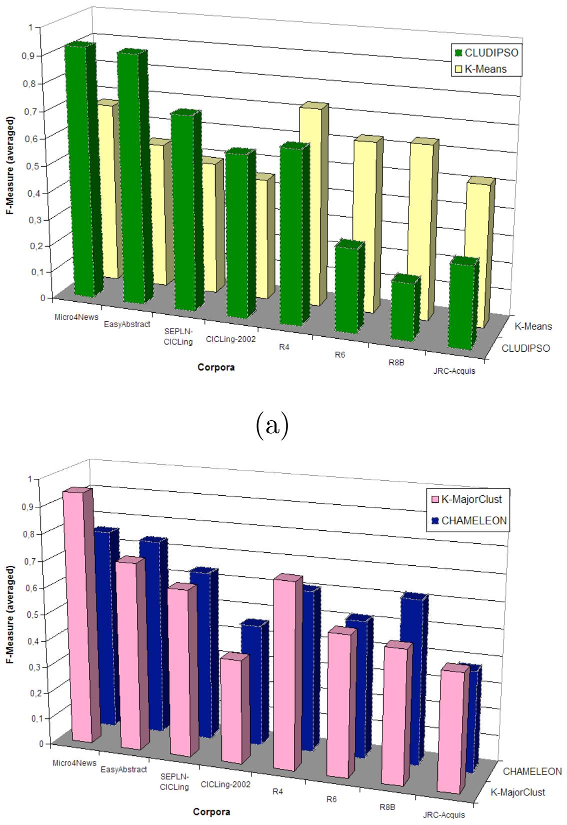
[1] CHAMELEON is not considered in the boxplots for *R4* and *R6* corpora because it obtains lower number of results making its distribution not comparable (from a statistical point of view) with the other algorithms.

With *R6*, a corpus bigger than *R4*, the Figure 2 shows that CLUDIPSO gets similar results to K-MajorClust, with similar median values and low dispersions in their boxplots. The boxplot of K-Means shows the best median value (around 0.6) but with a higher dispersion of values and larger whiskers, indicating that the distribution of values is slightly bigger (many different values) than that of CLUDIPSO. Again for this corpus, K-Means outperforms CLUDIPSO but the differences in favor of K-Means tend to increase with respect to the previous corpus (*R4*).

Figure 2 shows that, for the *R8B* corpus, CLUDIPSO and K-MajorClust have a low performance (median values around 0.25) but they have a lower dispersion than K-Means and CHAMELEON. The last one have a larger variability of results compared with that of K-Means (that is, larger whiskers). K-MajorClust and CLUDIPSO have less variability of F-measure values for that their boxplots do not show any whisker. Obviously, the best median values of K-Means and CHAMELEON are conclusive about a better performance of these algorithms with respect to CLUDIPSO.

The boxplot corresponding to *JRC-Acquis* corpus in Figure 2 shows that CLUDIPSO obtains the lowest results with a median value around 0.25 although it has minimum dispersion and the whiskers are the shortest compared with those of the remaining algorithms. For this corpus, the best median value (and best performance) was obtained by K-Means although with larger whiskers and larger dispersion.

Figure 3 shows the bars corresponding to the F-measure values averaged over the 50 experiments, obtained by each algorithm under study with each corpus. CLUDIPSO averaged F-measure bar (Figure 3 (a)) is the best with respect to those of the remaining algorithms for the four small short-text corpora (*Micro4News*, *EasyAbstracts*, *SEPLN-CICLing* and *CICLing-2002*). For the larger corpora (*R4*, *R6*, *R8B* and *JRC-Acquis*), CLUDIPSO obtained the worst averaged F-measure, except for *R4* for which CHAMELEON obtained the worst performance (Figure 3 (b)). That means that CLUDIPSO obtains best F-measure values (in average) for the small corpora. The deterioration of CLUDIPSO observed in the corresponding bar shows the low performance of this algorithm for larger corpora, in fact, the lowest averaged F-measure value was obtained with *R8B* which is the biggest corpus.



(a)

(b)

Figure 3: Performance deterioration of the algorithms on larger corpora.

As final conclusion of this statistical distribution study, it is possible to state that, according the search space grows (the number of documents increases), CLUDIPSO can not converge into good quality results even though it can still be considered a "robust" algorithm observing the dispersion of its results with small and larger corpora.

4.2 Discussion about the results

The clear difference of CLUDIPSO performance in both kinds of corpora (that is, with few and many documents) is probably due to the difficulty of the algorithm to explore the big search space that larger corpora imply: a larger number of documents implies a larger space to explore for a good clustering. This could be observed in the little improvement of performance reached during a single execution of CLUDIPSO when it had to evolve the big size particles (one dimension for each document) for the larger corpora. Additionally, the mechanism used to update the particles (proposed for discrete versions of PSO in (Ingaramo et al., 2009)) causes a slow search space exploration making the algorithm unable to find good solutions in a considerable number of cycles (that is 10,000). This slow exploration can be observed when CLUDIPSO finishes the execution of an experiment and the last performance improvement is obtained in the last iteration of the algorithm. However, to consider more than 10,000 cycles could be inadequate because the execution time spent for a CLUDIPSO single experiment could be notably increased.

5 Conclusions and Future Work

This work presented a study of performance of CLUDIPSO, a PSO-based clustering algorithm, when clustering short-text corpora. Previous results obtained by CLUDIPSO indicate that the approach is a highly competitive alternative to solve small short-text corpora, that is, corpora with no more than 50 documents. In this work, CLUDIPSO was also tested with larger size corpora and the performance was not comparable with other traditional algorithms like K-Means. In these comparisons, a constant deterioration of the F-measure values obtained with CLUDIPSO was observed while the number of documents in the corpora was increased. Some possible reasons about the poor performance, aim to the low exploration of the search space that CLUDIPSO makes with larger corpora due the large size of the particles. Additionally, the mechanism used to update the position of the particles causes very few improvements in the performance when the algorithm is executed a considerable number of cycles (more than 10,000).

Current works include the modification in the representation of the particles to consider sub-groups of documents instead of single documents in each dimension in order to reduce the length of the particles. The adaptation of several stages of the CLUDIPSO algorithm to incorporate information about the clustering problem itself in order to reduce the execution time spend by CLUDIPSO to process a single particle, will be considered.

Future works include improvements in the mechanism to update the particles in order to accelerate the exploration of the search space and, a study about the reliability of the results obtained by CLUDIPSO in comparison with those of algorithms representative of the state of the art.

Acknowledgements

This research work was done in the framework of Marie Curie actions PEOPLE-IRSES 269180 WiQ-Ei: Web information Quality Evaluation initiative. The work of the first author is partially funded by the UPV program PAID-02-10-2257. The work of the last two authors is partially funded by the MICINN research project TEXT-ENTERPRISE 2.0 TIN2009-13391-C04-03 (Plan I+D+i). The fourth author belongs to the VLC/CAMPUS Microcluster on Multimodal Interaction in Intelligent Systems.

References

Alexandrov, M., A. Gelbukh, and P. Rosso. 2005. An approach to clustering abstracts. In A. Montoyo, R. Muñoz, and E. Métais, editors, *Natural Language Processing and Information Systems*, volume 3513 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, pages 1–10.

Cagnina, L., S. Esquivel, and R. Gallard. 2004. Particle swarm optimization for sequencing problems: a case study. *Congress on Evolutionary Computation*, pages 536–541.

Eberhart, R. and J. Kennedy. 1995. A new optimizer using particle swarm theory. In *Proc. of the Sixth International Symposium on Micro Machine and Human Science, MHS'95*, pages 39–43, Nagoya, Japan.

Errecalde, M. and D. Ingaramo. 2008. Short-text corpora for clustering evaluation. Technical report, LIDIC.

Frank, A. and A. Asuncion. 2010. UCI machine learning repository.

Hu, X., R. Eberhart, and Y. Shi. 2003. Swarm intelligence for permutation optimization: a case study on n-queens problem. In *Proc. of the IEEE Swarm Intelligence Symposium*, pages 243–246.

Ingaramo, D., M. Errecalde, L. Cagnina, and P. Rosso, 2009. *Computational Intelli-*

gence and Bioengineering, chapter Particle Swarm Optimization for clustering short-text corpora, pages 3–19. IOS press.

Ingaramo, D., David Pinto, P. Rosso, and M. Errecalde. 2008. Evaluation of internal validity measures in short-text corpora. In *Proc. of the International Conference on Intelligent Text Processing and Computational Linguistics, CICLing 2008*, volume 4919 of *Lecture Notes in Computer Science*, pages 555–567. Springer-Verlag.

Karypis, G., E. Han, and V. Kumar. 1999. Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, 32:68–75.

MacQueen, J. B. 1967. Some methods for classification and analysis of multivariate observations. In *Proc. of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297.

Makagonov, P., M. Alexandrov, and A. Gelbukh. 2004. Clustering abstracts instead of full texts. In *Proc. of International Conference on Text, Speech and Dialogue, TSD 2004*, volume 3206 of *Lecture Notes in Artificial Intelligence*, pages 129–135.

Pinto, D., J. M. Benedí, and P. Rosso. 2007. Clustering narrow-domain short texts by using the Kullback-Leibler distance. In *Proc. of the International Conference on Intelligent Text Processing and Computational Linguistics, CICLing 2007*, volume 4394 of *Lecture Notes in Computer Science*, pages 611–622. Springer-Verlag.

Shi, Y. and R. Eberhart. 1998. A modified particle swarm optimizer. In *Proc. of the IEEE International Conference on Evolutionary Computation*, pages 69–73.

Stein, B. and O. Niggemann. 1999. On the nature of structure and its identification. In *Proc. of the 25th International Workshop on Graph Theoretic Concepts in Computer Science*, volume 1665 of *Lecture Notes in Computer Science*, pages 122–134. Springer-Verlag.

Steinberger, R., B. Pouliquen, A. Widiger, C. Ignat, T. Erjavec, D. Tufis, and D. Varga. 2006. The jrc-acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proc. of the 5th International Conference on Language Resources and Evaluation (LREC'2006)*, pages 24–26.

Tukey, J. W. 1977. *Exploratory data analysis.* Addison-Wesley Publishing Company, Reading, MA.