Procesamiento del lenguaje Natural, nº 41 (2008), pp. 81-87 recibido 7-05-2008; aceptado 16-06-2008

Density-based clustering of short-text corpora*

Agupamiento de textos cortos basado en densidad

Diego A. Ingaramo, Marcelo L. Errecalde
LIDIC, UNSL, San Luis, Argentina
Avda. Ejército de los Andes 950
{daingara,merreca}@unsl.edu.ar

Paolo Rosso
NLE Lab., DSIC, UPV, España
Camino de Vera s/n 46022
prosso@dsic.upv.es

Resumen: En este trabajo investigamos el desempeño de diferentes algoritmos de agrupamiento basados en densidad en colecciones de textos cortos y textos cortos de dominios restringidos. Nuestro objetivo es analizar en que medida las características de este tipo de colecciones impacta en el cálculo de la densidad de los agrupamientos y cuan robustos son este tipo de algoritmos a los distintos niveles de complejidad.
Palabras clave: agupamiento de textos cortos, algoritmos basados en densidad.

Abstract: In this work, we analyse the performance of different density-based algorithms on short-text and narrow domain short-text corpora. We attempt to determine to what extent the features of this kind of corpora impact on the density computation of the clusterings obtained and how robust these algorithms to the different complexity levels are.
Keywords: short-text clustering, density-based algorithms.

1 Introduction

In realistic document clustering problems, results cannot usually be evaluated with typical *external* measures like F-Measure, because the correct categorizations specified by a human editor are not available. Therefore, the quality of the resulting groups is evaluated with respect to *structural* properties expressed in different *Internal Clustering Validity Measures* (ICVM). Classical ICVM used as cluster validity measures include the Dunn and Davies-Bouldin indexes and new graph-based measures like *Density Expected Measure* (DEM) and λ-Measure (Stein, Meyer zu Eissen, and Wißbrock, 2003).[1] A central aspect to be considered in these cases is which are the ICVM that show an adequate *correlation* degree with the categorization criteria of a human editor.

In recent works (Stein, Meyer zu Eissen, and Wißbrock, 2003; Ingaramo et al., 2008), *density-based* ICVM like DEM have obtained the best correlation values with respect to the (external) F-measure, outperforming other more popular ICVM in experiments with samples of the RCV1 Reuters collection (Stein, Meyer zu Eissen, and Wißbrock, 2003) and short-text corpora (Ingaramo et al., 2008). According to these results, an algorithm which has a tendency to produce groupings with high density values, would achieve results that satisfies the information need of users.

Density-based algorithms are supposed to exhibit that tendency and they will be our main focus of attention in the present work. We are interested in testing these algorithms in problems with different degrees of complexity and in particular in collections containing *very short* texts, where additional difficulties are introduced due to the low frequencies of the document terms. This problem and other features, such as a high level of vocabulary overlapping among the categories of a corpus, can negatively affect the computation of the similarity between documents and the density of the document clusterings.

Work on "short-text clustering" is relevant, particularly if we consider the current/future mode for people to use 'small-language', e.g. blogs, text-messaging, snippets, etc. Potential applications in different areas of natural language processing may include re-ranking of snippets in information retrieval, and automatic clustering of scientific texts available on the Web (Alexandrov, Gelbukh, and Rosso, 2005).

In order to obtain a better understanding

* This work has been partially supported by the MCyT TIN2006-15265-C06-04 project, the ANPCyT and the Universidad Nacional de San Luis.

[1] See (Ingaramo et al., 2008) and (Stein, Meyer zu Eissen, and Wißbrock, 2003) for more detailed descriptions of these ICVM.

ISSN: 1135-5948

of the adequacy of density-based clustering algorithms for clustering short-text corpora, a deeper analysis of the relation between the features and difficulties of these corpora and the performance of different density-based algorithms is required. Specifically, we are interested in answering the following questions:

1. how a low frequency of words and the vocabulary overlapping affect the similarity estimation and the density of clustering?
2. which density algorithms are robust to these features?
3. how dense the results obtained by density-based algorithms are when applied to short-text collections? Are good these results from a user viewpoint?

To answer these questions we will use three popular density-based algorithms: *MajorClust*, *DBscan* and *Chameleon*. They will be tested on two different very short-text corpora which differ in the overlapping degree of their vocabularies. Results are also compared with a corpus which contains longer documents on well differentiated topics. In a nutshell, we want to consider situations where these algorithms have to deal with different level of complexity in the document collections under consideration. This complexity (or hardness) will be estimated with respect to the DEM value of the "correct" clustering.

The remainder of the paper is organized as follows. Section 2 presents our criteria for determining the hardness of short-text corpora from a density perspective; here, we also analyse the corpora that will be used in the experiments according to these criteria. The experimental results are shown in Section 3. Finally, some general conclusions are drawn and possible future work is discussed.

2 Density Estimation and Complexity of Short-text Corpora

In order to analyse the performance of different density-based algorithms we have to consider how they work in corpora with different levels of complexity. The term "hardness" has been recently used in previous works (Pinto and Rosso, 2007; Errecalde, Ingaramo, and Rosso, 2008) to refer to the complexity that a given corpus presents for clustering problems. This hardness is estimated in (Pinto and Rosso, 2007) considering the vocabulary overlapping among the categories of a corpus and in (Errecalde, Ingaramo, and Rosso, 2008) is determined with respect to the difficulty level that it presents for establishing an *accurate similarity measure* among its documents.

In the present work we will take a different perspective and we will focus on the density of the "correct" clustering defined by a human editor for estimating how complex a corpus is for a density-based algorithm. The rationale behind this idea is simple: if it is hard to identify dense groups in a document grouping defined by an expert, this collection will be similarly difficult for those clustering algorithms that search regions of high density. We will consider three corpora which are assumed to have different levels of difficulty from this perspective: in particular, we are interested in detecting how well the different density-based algorithms work with short-text corpora and narrow domain short-text corpora with respect to other more standard corpora. These corpora are introduced in the following subsection.

2.1 Data Sets

The complexity of clustering problems with short-text corpora demands a meticulous analysis of the features of each collection used in the experiments. For this reason, we will focus on specific characteristics of the collections such as document lengths and its closeness with respect to the topics considered in these documents. We attempt with this decision to avoid introducing other factors that can make the results incomparable.

With the exception of the CICling-2002 collection which has already been used in previous works (Makagonov, Alexandrov, and Gelbukh, 2004; Alexandrov, Gelbukh, and Rosso, 2005; Pinto, Benedí, and Rosso, 2007), the remaining two corpora were artificially generated with the goal of obtaining corpora with different levels of complexity with respect to the length of documents and vocabulary overlapping. Our intention was that in each corpus the similarity measure used to quantify the "closeness" between documents has different levels of complexity for detecting the conceptual proximity between two texts. In that way, the accuracy of the similarity measure will be different for the different collections and this fact will also

affect the density estimation of the document clusterings. However, other features such as the number of groups and number of documents per group were maintained the same for all collections in order to obtain comparable results.

It could be argued that our analysis is limited to small size collections. However, we believe that short-text clustering in general and clustering of narrow domain abstracts in particular, demand a detailed understanding of each collection that would be difficult to achieve with large size standard corpora.

In the following subsections, a general description of two collections used in this work is presented. These collections are introduced in increasing order of complexity. We begin with the Micro4News corpus, a collection of medium-length documents about well differentiated topics (low complexity). Then, the EasyAbstracts corpus with short-length documents (scientific abstracts) and well differentiated topics is presented (medium complexity corpus). These two new collections were created with similar general characteristics (number of groups and number of documents per group).[2] The CICling-2002 corpus with relatively high complexity was also used in our work. This collection is considered to be harder to cluster than the previous corpora since its documents are narrow domain abstracts (see (Pinto, Benedí, and Rosso, 2007) for a more detailed description of the corpus).

2.1.1 The Micro4News Corpus

This first collection was constructed with medium-length documents that correspond to four very different topics. Consequently, in this case it is supposed that the similarity measure will not have any problem in determining if two documents are semantically related. Its documents are significatively larger than CICling-2002 and talk about well differentiated topics. Documents were selected from four very different groups of the popular 20Newsgroups corpus (Lang, 1993): 1) *sci.med*, 2) *soc.religion.christian*, 3) *rec.autos* and 4) *comp.os.ms-windows.misc*. For each topic, the largest documents were selected. Thus, it was ensured that the average length of its documents were seven times (or more) the length of abstracts of the remaining two corpora.

2.1.2 The EasyAbstracts Corpus

This collection can be considered harder than the previous one because its documents are scientific abstracts (same characteristic as CICling-2002) and in consequence are short texts. It differs from CICling-2002 with respect to the overlapping degree of the documents' vocabulary. EasyAbstracts documents also refer to a shared thematic (*intelligent systems*) but its groups are not so closely related as the CICling-2002 groups are. EasyAbstracts was constructed with abstracts publicly available on Internet that correspond to articles of four international journals in the following fields: 1) *Machine Learning*, 2) *Heuristics in Optimization*, 3) *Automated reasoning* and 4) *Autonomous intelligent agents*. It is possible to select abstracts for these disciplines in a way that two abstracts of two different categories are not related at all. However, some degree of complexity can be introduced if abstracts of articles related to two or more EasyAbstracts's categories are used.[3] In the EasyAbstract corpus a few documents were included with these last features in order to increase the complexity respect to the Micro4News corpus. Nevertheless, the majority of documents in this collection clearly belong to a single group. This last fact allows us to assume that a similarity measure should not have any problem in representing the proximity among documents compared with the complexity of CICling2002 corpus.

2.2 Density of Clusterings

Our study of different density-based algorithms will take as reference the ICVM named *Density Expected Measure* and denoted usually as $\overline{\rho}$. This measure has shown in recent previous works (Stein, Meyer zu Eissen, and Wißbrock, 2003; Ingaramo et al., 2008) an interesting correlation with the (external) F-measure which is based on the information of a correct clustering specified by an expert. Following, some preliminary concepts and the definition of DEM are introduced.

Let us consider a data collection as a weighted graph $G = \langle V, E, w \rangle$ with node set

[2] A detailed description of the distribution and features of these two corpora is available in (Errecalde and Ingaramo, 2008) where you can also find the information on how to access the corpora for research purposes.

[3] For instance, abstracts which refer to *learning intelligent agents* or *agents with high level reasoning capabilities*.

V (representing documents), edge set E (representing similarity between documents) and weight function $w : E \rightarrow [0, 1]$ (representing a similarity function between documents).

A graph G $= \langle V, E, w\rangle$ may be called sparse if $|E| = \mathcal{O}(|V|)$, whereas it is called dense if $|E| = \mathcal{O}(|V|^2)$. Then we can compute the density θ of a graph from the equation $|E| = |V|^\theta$ where $w(G) = |V| + \sum_{e\in E} w(e)$, in the following manner:

$$w(G) = |V|^\theta \Leftrightarrow \theta = \frac{\ln(w(G))}{\ln(|V|)} \qquad (1)$$

θ can be used to compare the density of each induced subgraph $G^{'} = \langle V^{'}, E^{'}, w^{'}\rangle$ with respect to the density of the initial graph G. G' is sparse (dense) compared to G if $\frac{w(G^{'})}{|V'|^\theta}$ is smaller (bigger) than 1. Formally (Stein, Meyer zu Eissen, and Wißbrock, 2003), let $\mathcal{C} = \{C_1, .., C_k\}$ be a clustering of a weighted graph $G = \langle V, E, w\rangle$ and $G_i = \langle V_i, E_i, w_i\rangle$ be the induced subgraph of G with respect to cluster C_i. Then the *Density Expected Measure* $\overline{\rho}$ of a clustering $\mathcal{C}$ is obtained as shown in Eq. 2. A high value of $\overline{\rho}$ should indicate a good clustering.

$$\overline{\rho}(\mathcal{C}) = \sum_{i=1}^{k} \frac{|V_i|}{|V|} \cdot \frac{w(G_i)}{|V_i|^\theta} \qquad (2)$$

As can be observed, the $\overline{\rho}$ computation heavily depends on the similarity measure used for determining how close are two documents. Therefore, we should consider different similarity measures in order to observe which are the DEM values obtained in each case.

There are two main factors that usually impact on a similarity measure between documents: the document representation and the procedure used for computing the similarity between documents with this representation. One of the most widely used model for document representation is the *Vector Space Model* which has associated a family of weighting schemes that we will refer as the "SMART codifications" (Salton, 1971). Here, vector (document) similarity is usually measured by the *cosine* measure but other similarity measures derived from the Euclidean distance can also be used with this representation. Another popular document representation approach is the *set model* which considers a document as a set whose elements are the document's terms. In this case, proximity between documents is often quantified by set intersection ratios being the *Jaccard coefficient* one of the most popular scheme for measuring set similarity.

In our work, we used the Jaccard coefficient and the SMART system conventional code scheme with the cosine similarity measure. In the SMART system, each codification is composed by three letters: the first two letters refer, respectively, to the TF (Term Frequency) and IDF (Inverse Document Frequency) components, whereas the third one ($NORM$) indicates whether normalization is employed or not. Taking into account standard SMART nomenclature, we will consider five different alternatives for the TF component: n (natural), b (binary), l (logarithm), m (max-norm) and a (augnorm); two alternatives for the IDF component (n (none) and t) and two alternatives for normalization: n (no normalization) and c (cosine). In this way, a codification *ntc* will refer to the popular scheme where the weight for the i-th component of the vector for the document d is computed as $tf_{d,i} \times \log(\frac{N}{df_i})$ and then cosine normalization is applied. Here, N denotes the number of documents in the collection, $tf_{d,i}$ is the term frequency of the i-th term in the document d and df_i refers to the document frequency of i-th term over the collection (see (Manning and Schütze,) for a more detailed explanation). With this representation scheme we can generate 20 different codifications but we will only consider results with the 10 normalized codifications ("**c" codifications) because codifications without normalization give equivalent results when cosine similarity is used as proximity measure.

Previously it was explained that, in this work, we will use the density of the "correct" clustering defined by a human editor for estimating how complex a corpus for a density-based algorithm is. Table 1 presents these density values that correspond to the DEM values obtained with the correct clusterings of the three corpora, using in each case: a) SMART codifications and cosine similarity and, b) Jaccard Coefficient (denoted Jac).

Here, it can be observed that the traditional *ntc* codification with cosine similarity gives the highest values of DEM in each collection. In that sense, it should be noted that

the *mtc* codification is another valid candidate to be selected as the "best" codification. From now on, we will refer to the DEM value obtained with a correct clustering of a collection as the "*intrinsic*" DEM value of the collection. Obviously, different codifications and similarity measures used with a collection will produce different intrinsic DEM values.

Codif.	M4N	EasyAb	CiC02
atc	0.9	0.88	0.85
btc	0.9	0.88	0.84
mtc	**1.07**	**0.93**	**0.87**
ntc	**1.07**	**0.93**	**0.87**
Jac	0.78	0.74	0.79
anc	0.77	0.72	0.76
ltc	0.92	0.89	0.85
bnc	0.77	0.72	0.75
lnc	0.78	0.73	0.76
mnc	0.82	0.75	0.8
nnc	0.82	0.75	0.8

Table 1: "Intrinsic" density values

As can be observed in Table 1, the complexity of corpora directly impacts on the similarity measure and, in an indirect way, on the intrinsic DEM values obtained in each case. Considering the highest DEM values obtained with the *ntc* codification, it is evident that a very good value of DEM (1.07) is achieved for the Micro4News corpus (denoted M4N in the table). However, short-text collections exhibit decreasing intrinsic DEM values according to its complexity: 0.93 for the EasyAbstracts collection and the lowest value of DEM (0.87) for the CICling2002 corpus.

It is important to note that the impact of the hardness of corpora on the similarity measure can also be appreciated in the results delivered by other internal validity measures on the "correct" clustering. For example, in Figure 1 the silhouette graphics (Rousseeuw, 1987) are shown for the best SMART codification (*ntc*) with similarity cosine for the three collections we will use in our study. In the first case (Micro4News), each document shows an evident membership degree to its group but results with EasyAbstracts are not so good and in the CiCLing2002 case, the silhouette graphics are definitively bad. These results, and the intrinsic DEM values obtained, are a clear evidence of the complexity of short-text and narrow domain short-text corpora for clustering purposes, with respect to standard document collections. In the next section, we will show how robust three density-based algorithms are to the difficulties that presents each collection.

3 Experimental results

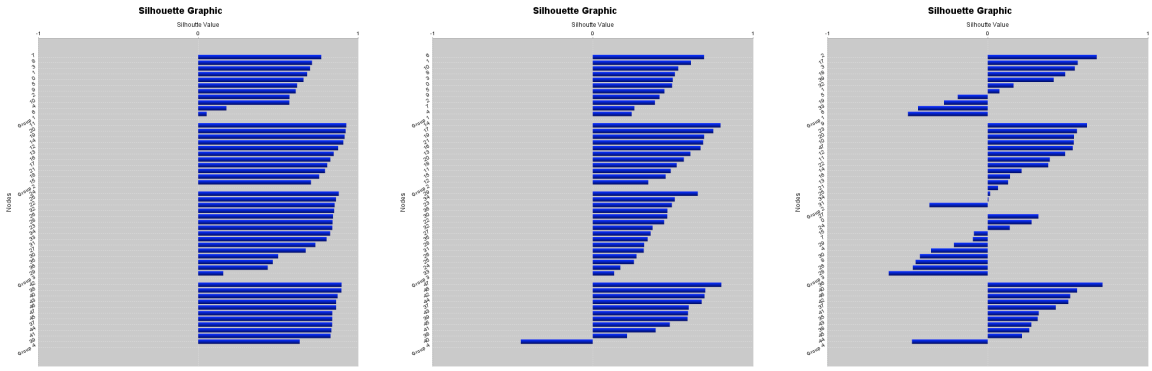
In this section, we will analyse the performance of the different density-based algorithms organizing the discussion around the results obtained with each collection. We consider for the experimentation the representation schemes that showed the highest intrinsic DEM values for each corpus (see Table 1). In consequence, the results presented below correspond to the *ntc* codification with cosine similarity for the three collections considered.

We used three algorithms which are considered in different works (Meyer zu Eissen, 2007; Stein and Busch, 2005) as representative of the density-based approach to the clustering problem: MajorClust (Stein and Niggemann, 1999), DBSCAN (Ester et al., 1996) and Chameleon (Karypis, Han, and Vipin, 1999)[4]. Basically, these algorithms attempt to separate the set of objects (documents) into subsets of similar densities. However, a significative difference between them is whether the algorithm requires information about the correct number of groups (k) or not. This information has to be provided to the Chameleon algorithm but MajorClust and DBSCAN determine the cluster's number k automatically. Space limitations prevent us from giving a more detailed explanation of the algorithms, but the interested reader can obtain more information in (Stein and Niggemann, 1999; Ester et al., 1996; Karypis, Han, and Vipin, 1999).

3.1 Micro4News

In Table 2, we can observe that in this corpus MajorClust obtains the highest DEM values. Another interesting aspect observed during the experimentation is that despite considering different parameters that influence the way the algorithm obtains the results (threshold values), MajorClust usually yield similar (or the same) results with density values in the interval [1.05 : 1.1]. Only 6 different results were obtained taking different threshold values and 5 of them had DEM values greater than the intrinsic DEM

[4]We indeed use a variant of Chameleon provided in the CLUTO toolkit: www.cs.umn.edu/ karypis/cluto.



(Micro4News) **(EasyAbstracts)** **(CiCLing2002)**

Figure 1: Silhouette graphics.

of the collection (1.07). The F-measure values corresponding to these five DEM values also were significatively high (in the interval [0.88 : 0.96]). On the other hand, Chameleon obtained lower DEM values and with more variance than the remainder algorithms. However, it produced clusterings with F-measure values as good as the results obtained by MajorClust (0.96). The DEM values obtained with DBSCAN are higher than the values obtained with Chameleon and lower than the values corresponding to MajorClust. Nonetheless, respect to its F_{max} value, this value is lower than the values obtained with the other algorithms.

3.2 EasyAbstract

The results corresponding to this collection (Table 3), confirm the tendency previously exhibited by MajorClust to obtain groupings with the highest DEM values.[5] However, it is important to observe that in this case the differences with the values obtained with the other algorithms are small, giving Chameleon and DBSCAN very similar results. Considering the F-measure values, we can see that the highest F_{max} is obtained by MajorClust (0.98), a similar performance of Chameleon (0.96) and a poor functioning of DBSCAN (0.72). If only this information is considered, these results, could be interpreted as a better performance of MajorClust with respect to Chameleon. However, it is important to take also into account the F_{avg} and F_{min} values where Chameleon clearly outperforms MajorClust. This assertion can be graphically appreciated in Figure 2 which shows DEM values vs F-measure values of groupings obtained with MajorClust(left) and Chameleon(right). We can observe that MajorClust achieves the highest F-measure value (0.98) but the remaining values exhibit a great variation and oscillate in the interval $[0.44 - 0.96]$. Furthermore, a weak correlation can be observed between the DEM values and the corresponding F-measure values. Chameleon obtains in this case very different (and interesting) results. We can observe that a small number of results were obtained. However, a considerable proportion of them reached F-measure values greater than 0.82. For this algorithm, also is evident the good correlation between the DEM values and the F-measure values.

These differences in the results of both algorithms, with respect to the correlations between DEM and F-measure values, require a deeper and more detailed analysis. In this case it may be useful to consider the values corresponding to the Spearman rank correlation index (Myers and Well, 2002), that are shown in Table 4 for each collection and algorithm used in the experiments. Here, we can note that Chameleon shows the best correlations between DEM and F-measure values for all the collections considered. These results are indicative that, in the case of using EasyAbstract, with an algorithm like Chameleon we can expect that results with high DEM values correspond to high F-measure values. This performance cannot be guaranteed with MajorClust which exhibits the worst correlation value (0.15). In order to understand the causes for this poor performance of MajorClust, we analysed the groupings produced by this algorithm. An important aspect observed was that only 15% of the results had 4 groups (the correct num-

[5] All the DEM values obtained were higher than the intrinsic DEM of the collection (0.93).

Algorithm	DEM_{avg}	DEM_{min}	DEM_{max}	F_{avg}	F_{min}	F_{max}
MajorClust	1.08	1.05	1.1	0.90	0.76	0.96
Chameleon	1.03	0.97	1.07	0.76	0.46	0.96
DBSCAN	1.05	1.01	1.1	0.82	0.71	0.88

Table 2: Micro4News: results with different density-based algorithms

Algorithm	DEM_{avg}	DEM_{min}	DEM_{max}	F_{avg}	F_{min}	F_{max}
MajorClust	0.94	0.93	0.96	0.69	0.44	0.98
Chameleon	0.93	0.92	0.94	0.84	0.66	0.96
DBSCAN	0.93	0.91	0.94	0.66	0.62	0.72

Table 3: EasyAbstract: results with different density-based algorithms

ber of groups of the collection). This information is indicative: in those collections where the intrinsic DEM is not so high (as in the previous collection), MajorClust will have problems for generating groupings with the correct number of clusters. Therefore, it will have little chances of producing a result with a high F-measure value. As an argument in favor of MajorClust, we can say that for those cases where the results had the correct number of clusters (4), the F-measure values achieved for MajorClust were comparable to those obtained with Chameleon.

3.3 CICLing2002

In this collection it is evident that the similarity measure does not adequately reflects the conceptual proximity between documents and, therefore, the DEM values are not reliable indicators of the quality of results. An important consequence of this fact, is that the algorithms which explicitly attempt to achieve high values of this ICVM cannot offer any guarantee about obtaining good results from the user viewpoint. This affirmation can be verified in the results shown in Table 5 where high values of DEM[6] do not correspond to high F-measure values which are, in general, very low. Nonetheless, we can observe an interesting result in this last experiments. Chameleon reaches, as in the previous collection, better F-measure values than the other algorithms considered and also shows the best Spearman correlation value. Based on this information we can conclude that Chameleon does not always reach the highest DEM values but its correlation values between the density of the clustering obtained and the F-measure values are very good in collections with diverse complexity levels. An important additional observation is that Chameleon achieved, in all the collections considered, the highest (or near to the highest) F-measure value. This suggest that the mechanisms used in this algorithm for clustering the documents usually agree with the grouping criteria of a human expert and it deserves additional research work.

[6]Greater than 0.87, the intrinsic DEM of this collection.

4 Conclusions and Future Work

In this research work we investigated the relations between the hardness of short-text corpora, the density expected measure and the robustness of density-based algorithms. Our first conclusion is that in collections with medium-length documents with groups that correspond to very different topics, we probably observe high intrinsic DEM values. In these cases the three density-based algorithms will usually be able to reach these high density values in the results obtained and will also obtain good F-measure values.

In short text corpora, their intrinsic DEM is negatively affected by the low frequencies of the document terms. This negative influence is incremented when narrow domains are involved. In these situations, all the algorithms were affected but Chameleon showed a very interesting correlation level between the DEM values and the F-measure values. These strengths of Chameleon combined with the good results of F-measure deserve further research work employing this clustering algorithm.

With respect to the density values of the results, MajorClust reached the highest DEM values and sometimes it also obtained good F-measure values. However, in collections with low intrinsic DEM values it generated a considerable number of results with a wrong

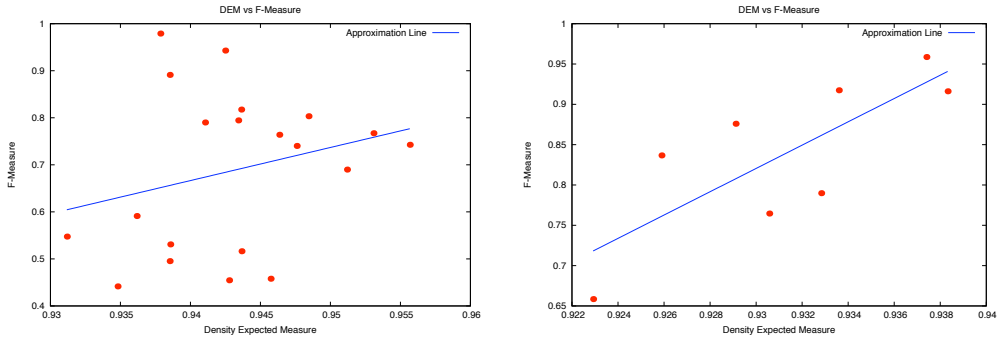


Figure 2: EasyAbstracts: DEM vs F-measure for MajorClust(left) and Chameleon(right).

Algorithm	**4MNG**	**EasyAbstract**	**CICLing2002**
MajorClust	-0.08	0.15	0.13
Chameleon	0.88	0.74	0.32
DBSCAN	0.87	0.5	-0.24

Table 4: Spearman Correlation between DEM and F-Measure

Algorithm	DEM_{avg}	DEM_{min}	DEM_{max}	F_{avg}	F_{min}	F_{max}
MajorClust	0.92	0.91	0.94	0.43	0.37	0.58
Chameleon	0.91	0.9	0.93	0.55	0.5	0.66
DBSCAN	0.91	0.88	0.95	0.47	0.42	0.56

Table 5: CICLing2002: results with different density-based algorithms

number of groups, affecting in that way the F-measure values obtained.

References

Alexandrov, M., A. Gelbukh, and P. Rosso. 2005. An approach to clustering abstracts. In *Proc. of NLDB-05*, volume 3513 of *LNCS*, pages 8–13.

Errecalde, M. and D. Ingaramo. 2008. Short-text corpora for clustering evaluation. http://www.dirinfo.unsl.edu.ar/∼ia/resources/shortexts.pdf. Technical report, LIDIC.

Errecalde, M., D. Ingaramo, and P. Rosso. 2008. Proximity estimation and hardness of short-text corpora. In *TIR-08 (to appear)*.

Ester, M., H. Kriegel, J. Sander, and X. Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. of KDD-96*, pages 226–231.

Ingaramo, D., D. Pinto, P. Rosso, and M. Errecalde. 2008. Evaluation of internal validity measures in short-text corpora. In *Proc. of CICLing 2008*, volume 4919 of *LNCS*, pages 555–567.

Karypis, G., E.-H. Han, and K. Vipin. 1999. Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, 32(8):68–75.

Lang, K. 1993. 20 newsgroups, the original data set. http://kdd.ics.uci.edu/databases/ 20newsgroups/20newsgroups.html.

Makagonov, P., M. Alexandrov, and A. Gelbukh. 2004. Clustering abstracts instead of full texts. In *Proc. of the TSD-2004*, volume 3206 of *LNAI*, pages 129–135.

Manning, C. D. and H. Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press.

Meyer zu Eissen, S. 2007. *On Information Need and Categorizing Search*. Dissertation, University of Paderborn, Feb.

Myers, J. and A. Well. 2002. *Research Design and Statistical Analysis*. Lawrence Erlbaum Associates, second edition.

Pinto, D., J. M. Benedí, and P. Rosso. 2007. Clustering narrow-domain short texts by using the Kullback-Leibler distance. In *Proc. of CICLing 2007*, volume 4394 of *LNCS*, pages 611–622.

Pinto, D. and P. Rosso. 2007. On the relative hardness of clustering corpora. In *Proc. of TSD07*, volume 4629 of *LNAI*, pages 155–161.

Rousseeuw, P. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.*, 20(1):53–65.

Salton, G. 1971. *The Smart Retrieval System: Experiments in Automatic Document Processing*. Prentice Hall.

Stein, B. and M. Busch. 2005. Density-based Cluster Algorithms in Low-dimensional and High-dimensional Applications. In *TIR 05*, pages 45–56.

Stein, B., S. Meyer zu Eissen, and F. Wißbrock. 2003. On cluster validity and the information need of users. In *3rd IASTED*, pages 216–221.

Stein, B. and O. Niggemann. 1999. On the Nature of Structure and its Identification. In *Proc. of WG99*, volume 1665 of *LNCS*, pages 122–134.