A hybrid framework for news clustering based on the DBSCAN-Martingale and LDA

Ilias Gialampoukidis, Stefanos Vrochidis, and Ioannis Kompatsiaris

Information Technologies Institute, CERTH, Thessaloniki, Greece

Abstract. Nowadays there is an important need by journalists and media monitoring companies to cluster news in large amounts of web articles, in order to ensure fast access to their topics or events of interest. Our aim in this work is to identify groups of news articles that share a common topic or event, without a priori knowledge of the number of clusters. The estimation of the correct number of topics is a challenging issue, due to the existence of "noise", i.e. news articles which are irrelevant to all other topics. In this context, we introduce a novel density-based news clustering framework, in which the assignment of news articles to topics is done by the well-established Latent Dirichlet Allocation, but the estimation of the number of clusters is performed by our novel method, called "DBSCAN-Martingale", which allows for extracting noise from the dataset and progressively extracts clusters from an OPTICS reachability plot. We evaluate our framework and the DBSCAN-Martingale on the 20newsgroups-mini dataset and on 220 web news articles, which are references to specific Wikipedia pages. Among twenty methods for news clustering, without knowing the number of clusters k, the framework of DBSCAN-Martingale provides the correct number of clusters and the highest Normalized Mutual Information.

Keywords: Clustering News Articles, Latent Dirichlet Allocation, DBSCAN-Martingale

1 Introduction

Clustering news articles is a very important problem for journalists and media monitoring companies, because of their need to quickly detect interesting articles. This problem becomes also very challenging and complex, given the relatively large amount of news articles produced on a daily basis. The challenges of the aforementioned problem are summarized into two main directions: (a) discover the correct number of news clusters and (b) group the most similar news articles into news clusters. We face these challenges under the following assumptions. Firstly, we take into account that real data is highly noisy and the number of clusters is not known. Secondly, we assume that there is a lower bound for the minimum number of documents per cluster. Thirdly, we consider the names/labels of the clusters unknown.

Towards addressing this problem, we introduce a novel hybrid clustering framework for news clustering, which combines automatic estimation of the number of clusters and assignment of news articles into topics of interest. The estimation of the number of clusters is done by our novel "DBSCAN-Martingale", which can deal with the aforementioned assumptions. The main idea is to progressively extract all clusters (extracted by a density-based algorithm) by applying Doob's martingale and then apply a well-established method for the assignment of news articles to topics, such as Latent Dirichlet Allocation (LDA). The proposed hybrid framework does not consider known the number of news clusters, but requires only the more intuitive parameter *minPts*, as a lower bound for the number of documents per topic. Each realization of the DBSCAN-Martingale provides the number of detected topics and, due to randomness, this number is a random variable. As the final number of detected topics, we use the majority vote over 10 realizations of the DBSCAN-Martingale. Our contribution is summarized as follows:

- We present our novel DBSCAN-Martingale process, which progressively estimates the number of clusters in a dataset.
- We introduce a novel hybrid news clustering framework, which combines our DBSCAN-Martingale with Latent Dirichlet Allocation.

In the following, we present, in Section 2, existing approaches for news clustering and density-based clustering. In Section 3, we propose a new hybrid framework for news clustering, where the number of news clusters is estimated by our "DBSCAN-Martingale", which is presented in Section 4. Finally, in Section 5, we test both our novel method for estimating the number of clusters and our news clustering framework in four datasets of various sizes.

2 Related Work

News clustering is tackled as a text clustering problem [1], which usually involves feature selection [25], spectral clustering [21] and k-means oriented [1] techniques, assuming mainly that the number of news clusters is known. We consider the more general and realistic case, where the number of clusters is unknown and it is possible to have news articles which do not belong to any of the clusters.

Latent Dirichlet Allocation (LDA) [4] is a popular model for topic modeling, given the number of topics k. LDA has been generalized to nonparametric Bayesian approaches, such as the hierarchical Dirichlet process [29] and DP-means [20], which predict the number of topics k. The extraction of the correct number of topics is equivalent to the estimation of the correct number of clusters in a dataset. The majority vote among 30 clustering indices has recently been proposed in [7] as an indicator for the number of clusters in a dataset. In contrast, we propose an alternative majority vote among 10 realizations of the "DBSCAN-Martingale", which is a modification of the DBSCAN algorithm [12] and has three main advantages and characteristics: (a) they discover clusters with not-necessarily regular shapes, (b) they do not require the number of clusters and (c) they extract noise. The parameters of DBSCAN are the density level ϵ and a lower bound for the minimum number of points per cluster: $minPts$.

Other approaches for clustering that could be applied to news clustering, without knowing the number of clusters, are based on density based clustering algorithms. The graph-analogue of DBSCAN has been presented in [5] and dynamically adjusting the density level ϵ, the nested hierarchical sequence of clusterings results to the HDBSCAN algorithm [5]. OPTICS [2] allows for determining the number of clusters in a dataset by counting the "dents" of the OPTICS reachability plot. F-OPTICS [28] has reduced the computational cost of the OPTICS algorithm using a probabilistic approach of the reachability distance, without significant accuracy reduction. The OPTICS-ξ algorithm [2] requires an extra parameter ξ, which has to be manually set in order to find "dents" in the OPTICS reachability plot. The automatic extraction of clusters from the OPTICS reachability plot, as an extension of the OPTICS-ξ algorithm, has been presented in [27] and has been outperformed by HDBSCAN-EOM [5] in several datasets. We will examine whether some of these density based algorithms perform well on the news clustering problem and we shall compare them with our DBSCAN-Martingale, which is a modification of DBSCAN, where the density level ϵ is a random variable and the clusters are progressively extracted.

3 The DBSCAN-Martingale framework for news clustering

We propose a novel framework for news clustering, where the number of clusters k is estimated using the DBSCAN-Martingale and documents are assigned to k topics using Latent Dirichlet Allocation (LDA).

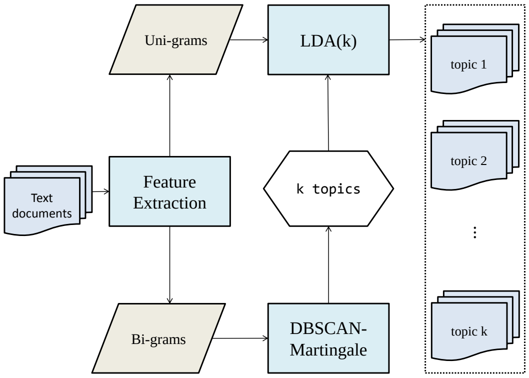


Fig. 1: Our hybrid framework for news clustering, using the DBSCAN-Martingale and Latent Dirichlet Allocation.

We combine DBSCAN and LDA because LDA performs well on text clustering but requires the number of clusters. On the other hand, density-based algorithms do not require the number of clusters, but their performance in text clustering is limited, when compared to LDA.

LDA [4] is a probabilistic topic model, which assumes a Bag-of-Words representation of the collection of documents. Each topic is a distribution over terms in a fixed vocabulary, which assigns probabilities to words. Moreover, LDA assumes that documents exhibit multiple topics and assigns a probability distribution on the set of documents. Finally, LDA assumes that the order of words does not matter and, therefore, is not applicable to word n-grams for $n \geq 2$.

We refer to word n-grams as "uni-grams" for $n = 1$ and as "bi-grams" for $n = 2$. The DBSCAN-Martingale performs well on the bi-grams, following the concept of "phrase extraction" [1]. We restrict our study on textual features (n-grams) in the present work and spatiotemporal features are not used.

In Figure 1 the estimation of the number of clusters is done by DBSCAN-Martingale and LDA follows for the assignment of text documents to clusters.

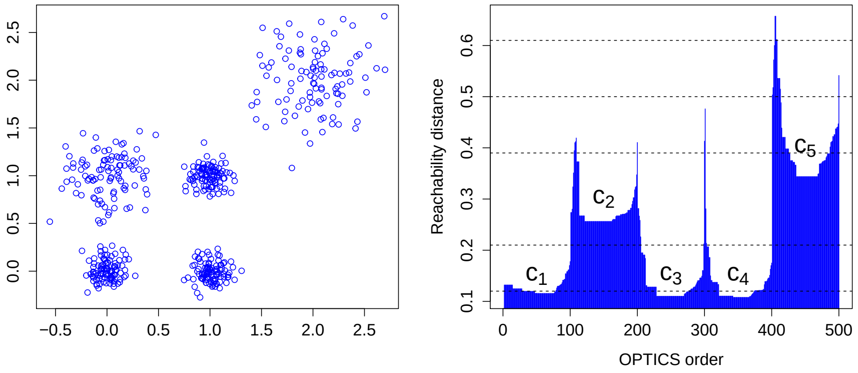
4 DBSCAN-Martingale

In this Section, we show the construction of the DBSCAN-Martingale. In Section 4.1 we provide the necessary background in density-based clustering and the notation which we adopt. In Section 4.2, we progressively estimate the number of clusters in a dataset by defining a stochastic process, which is then shown (Section 4.3) to be a Martingale process.

4.1 Notation and Preliminaries on DBSCAN

Given a dataset of n-points, density-based clustering algorithms provide as output the clustering vector C. Assuming there are k clusters in the dataset, some of the points are assigned to a cluster and some points do not belong to any of the k clusters. When a point $j = 1, 2, \ldots, n$ is assigned to one of the k clusters, the j-th element of the clustering vector C, denoted by $C[j]$ takes the value of the cluster ID from the set $\{1, 2, \ldots, k\}$. Otherwise, the j-th point does not belong to any cluster, it is marked as "noise" and the corresponding value in the clustering vector becomes zero, i.e. $C[j] = 0$. Therefore, the clustering vector C is a n-dimensional vector with values in $\{0, 1, 2, \ldots, k\}$.

The algorithm DBSCAN [12] is a density-based algorithm, which provides one clustering vector, given two parameters, the density level ϵ and the parameter $minPts$. We denote the clustering vector provided by the algorithm DBSCAN by $C_{DBSCAN(\epsilon,minPts)}$ or simply $C_{DBSCAN(\epsilon)}$ because the parameter $minPts$ is considered as a pre-defined fixed parameter. For low values of ϵ, $C_{DBSCAN(\epsilon)}$ is a vector of zeros (all points are marked as noise). On the other hand, for high values of ϵ, $C_{DBSCAN(\epsilon)}$ is a column vector of ones. Apparently, if a clustering vector has only zeros and ones, only one cluster has been detected and the partitioning is trivial.



(a) Data

(b) OPTICS clusters

Fig. 2: OPTICS reachability plot and randomly generated density levels

Clusters detected by DBSCAN strongly depend on the density level ϵ. An indicative example is shown in Figure 2(a), where the 5 clusters do not have the same density, and it is evident that there is no single value of ϵ that can output all the clusters. In Figure 2(b), we illustrate the corresponding OPTICS reachability plot with 5 randomly selected density levels (horizontal dashed lines) and none of them is able to extract all clusters $C_1, C_2, \ldots, C_5$ into one clustering vector C.

In order to deal with this problem we introduce (in Sections 4.2 and 4.3) an extension of DBSCAN based on Doob's Martingale, which allows for introducing a random variable ϵ and involves the construction of a Martingale process, which progressively approaches the clustering vector which contains all clusters so as to determine the number of clusters.

4.2 Estimation of the number of clusters with the DBSCAN-Martingale

We introduce a probabilistic method to estimate the number of clusters, by constructing a martingale stochastic process [11], which is able to progressively extract all clusters for all density levels. The martingale construction is, in general, based on Doob's martingale [11], in which we progressively gain knowledge about the result of a random variable. In the present work, the random variable that needs to be known is the vector of cluster IDs, which is a combination of T clustering vectors $C_{DBSCAN(\epsilon_t)}, t = 1, 2, \ldots, T$.

First, we generate a sample of size T with random numbers $\epsilon_t, t = 1, 2, \ldots, T$ uniformly in $[0, \epsilon_{max}]$, where ϵ_{max} is an upper bound for the density levels. The sample of $\epsilon_t, t = 1, 2, \ldots, T$ is sorted in increasing order and the values of ϵ_t

can be demonstrated on an OPTICS reachability plot, as shown in Figure 2 ($T = 5$). For each density level ϵ_t we find the corresponding clustering vectors $C_{DBSCAN(\epsilon_t)}$ for all stages $t = 1, 2, \ldots, T$.

In the beginning of the algorithm, there are no clusters detected. In the first stage ($t = 1$), all clusters detected by $C_{DBSCAN(\epsilon_1)}$ are kept, corresponding to the lowest density level ϵ_1. In the second stage ($t = 2$), some of the detected clusters by $C_{DBSCAN(\epsilon_2)}$ are new and some of them have also been detected at previous stage ($t = 1$). In order to keep only the newly detected clusters of the second stage ($t = 2$), we keep only groups of numbers of the same cluster ID with size greater than $minPts$.

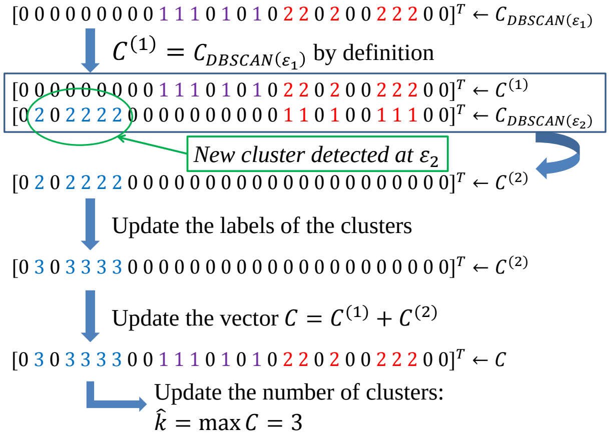

$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 2\ 2\ 0\ 2\ 0\ 0\ 2\ 2\ 2\ 0\ 0]^T \leftarrow C_{DBSCAN(\varepsilon_1)}$
$C^{(1)} = C_{DBSCAN(\varepsilon_1)}$ by definition
$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 2\ 2\ 0\ 2\ 0\ 0\ 2\ 2\ 2\ 0\ 0]^T \leftarrow C^{(1)}$
$[0\ 2\ 0\ 2\ 2\ 2\ 2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0]^T \leftarrow C_{DBSCAN(\varepsilon_2)}$
New cluster detected at ε_2
$[0\ 2\ 0\ 2\ 2\ 2\ 2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T \leftarrow C^{(2)}$
Update the labels of the clusters
$[0\ 3\ 0\ 3\ 3\ 3\ 3\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T \leftarrow C^{(2)}$
Update the vector $C = C^{(1)} + C^{(2)}$
$[0\ 3\ 0\ 3\ 3\ 3\ 3\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 2\ 2\ 0\ 2\ 0\ 0\ 2\ 2\ 2\ 0\ 0]^T \leftarrow C$
Update the number of clusters:
$\hat{k} = \max C = 3$

Fig. 3: One realization of the DBSCAN-Martingale with $T = 2$ iterations. The points with cluster label **2** in $C^{(1)}$ are re-discovered as a cluster by $C_{DBSCAN(\epsilon_2)}$ but the update rule keeps only the newly detected cluster.

Formally, we define the sequence of vectors $C^{(t)}, t = 1, 2, \ldots, T$, where $C^{(1)} = C_{DBSCAN(\epsilon_1)}$ and:

$$C^{(t)}[j] := \begin{cases} 0 & \text{if point } j \text{ belongs to a previously extracted cluster} \\ C_{DBSCAN(\epsilon_t)}[j] & \text{otherwise} \end{cases} \tag{1}$$

Since the stochastic process $C^{(t)}, t = 1, 2, \ldots, T$ is a martingale, as shown in Section 4.3, and $C_{DBSCAN(\epsilon_t)}$ is the output of DBSCAN for the density level ϵ_t, the proposed method is called "DBSCAN-Martingale".

Finally, we relabel the cluster IDs. Assuming that r clusters have been detected for the first time at stage t, we update the cluster labels of $C^{(t)}$ starting from $1 + \max_j C^{(t-1)}[j]$ to $r + \max_j C^{(t-1)}[j]$. Note that the maximum value of a clustering vector coincides with the number of clusters.

The sum of all vectors $C^{(t)}$ up to stage T is the final clustering vector of our algorithm:

$$C = C^{(1)} + C^{(2)} + \cdots C^{(T)} \tag{2}$$

The estimated number of clusters $\hat{k}$ is the maximum value of the final clustering vector C:

$$\hat{k} = \max_{j} C[j] \tag{3}$$

In Figure 3, we adopt the notation X^T for the transpose of the matrix or vector X, in order to demonstrate the estimation of the number of clusters after two iterations of the DBSCAN-Martingale.

The process we have formulated, namely the DBSCAN-Martingale, is represented as pseudo code in Algorithm 1. Algorithm 1 extracts clusters sequentially, combines them into one single clustering vector and outputs the most updated estimation of the number of clusters $\hat{k}$.

Algorithm 1: DBSCAN-Martingale($minPts$) **return** $\hat{k}$

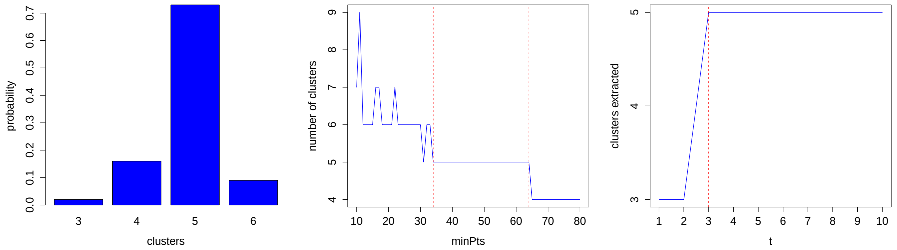
```
Generate a random sample of $T$ values in $[0, \epsilon_{max}]$
Sort the generated sample $\epsilon_t, t = 1, 2, \ldots, T$
for $t = 1$ to $T$
    find $C_{DBSCAN(\epsilon_t)}$
    compute $C^{(t)}$ as in Eq. (1)
    update the cluster IDs
    update the vector $C$ as in Eq. (2)
    update $\hat{k} = \max_j C[j]$
end for
return $\hat{k}$
```

The DBSCAN-Martingale requires T iterations of the DBSCAN algorithm, which runs in $\mathcal{O}(n \log n)$ if a tree-based spatial index can be used and in $\mathcal{O}(n^2)$ without tree-based spatial indexing [2]. Therefore, the DBSCAN-Martingale runs in $\mathcal{O}(Tn \log n)$ for tree-based indexed datasets and in $\mathcal{O}(Tn^2)$ without tree-based indexing. Our code is written in R[1], using the dbscan[2] package, which runs DBSCAN in $\mathcal{O}(n \log n)$ with kd-tree data structures for fast nearest neighbor search.

The DBSCAN-Martingale (one execution of Algorithm 1) is illustrated, for example, on the OPTICS reachability plot of Figure 2 (b) where, for the random sample of density levels $\epsilon_t, t = 1, 2, \ldots, 5$ (horizontal dashed lines), we sequentially extract all clusters. In the first density level $\epsilon_1 = 0.12$, DBSCAN-Martingale extracts the clusters C_1, C_3 and C_4, but in the density level $\epsilon_2 = 0.21$

[1] https://www.r-project.org/

[2] https://cran.r-project.org/web/packages/dbscan/index.html



(a) number of clusters

(b) robustness to the parameter *minPts*

(c) Convergence to k (one realization)

Fig. 4: The number of clusters as generated by DBSCAN-Martingale ($minPts = 50$) after 100 realizations

```
Algorithm 2: MajorityVote($realizations, minPts$) return $\hat{k}$
$clusters = \emptyset$, $k = 0$
for $r = 1$ to $realizations$
    $k$ =DBSCAN-Martingale($minPts$)
    $clusters$ = AppendTo($clusters, k$)
end for
$\hat{k}$ = mode($clusters$)
return $\hat{k}$
```

no new clusters are extracted. In the third density level, $\epsilon_3 = 0.39$, the clusters C_2 and C_5 are added to the final clustering vector and in the other density levels, ϵ_4 and ϵ_5 there are no new clusters to extract. The number of clusters extracted up to stage t is shown in Figure 4(c). Observe that at $t = 3$ iterations, DBSCAN-Martingale has output $k = 5$ and for all iterations $t > 3$ there are no more clusters to extract. Increasing the total number of iterations T will needlessly introduce additional computational cost in the estimation of the number clusters $\hat{k}$.

The estimation of number of clusters $\hat{k}$ is a random variable, because it inherits the randomness of the density levels $\epsilon_t, t = 1, 2, \dots, T$. For each execution of Algorithm 1, one realization of the DBSCAN-Martingale generates $\hat{k}$, so we propose as the final estimation of the number of clusters the majority vote over 10 realizations of the DBSCAN-Martingale.

Algorithm 2 outputs the majority vote over a fixed number of realizations of the DBSCAN-Martingale. For each realization, the estimated number of clusters k is added to the list *clusters* and the majority vote is obtained from the mode of *clusters*, since the mode is defined as the most frequent value in a list. The percentage of realizations where the DBSCAN-Martingale outputs exactly $\hat{k}$ clusters is a probability distribution, such as the one shown in Figure 4(a), which corresponds to the illustrative dataset of Figure 2(a). Finally, we note that the same result ($\hat{k} = 5$) appears for a wide range of the parameter *minPts* (Figure 4(b)), a fact that demonstrates the robustness of our approach.

4.3 The sequence of vectors $C^{(t)}$ is a martingale process

Martingale is a random process $X_1, X_2, \ldots$ for which the expected future value of X_{t+1}, given all prior values $X_1, X_2, \ldots, X_t$, is equal to the present observed value X_t. Doob's martingale is a generic martingale construction, in which our knowledge about a random variable is progressively obtained:

Definition 1. *(Doob's Martingale) [11]. Let $X, Y_1, Y_2, \ldots$ be any random variables with $E[|X|] < \infty$. Then if X_t is defined by $X_t = E[X|Y_1, Y_2, \ldots, Y_t]$, the sequence of $X_t, t = 1, 2, \ldots$ is a martingale.*

In this context, we will show that the sequence of clustering vectors $X_t = C^{(1)} + C^{(2)} + \cdots + C^{(t)}, t = 1, 2, \ldots, T$ is Doob's martingale for the sequence of random variables $Y_t = C_{DBSCAN(\epsilon_t)}, t = 1, 2, \ldots, T$.

We denote by $< Z_i, Z_l >= \sum_j Z_i[j] \cdot Z_l[j]$ the inner product of any two vectors Z_i and Z_l and we prove the following Lemma:

Lemma 1. *If two clustering vectors Z_i, Z_l are mutually orthogonal, they contain different clusters.*

Proof. The values of the clustering vectors are cluster IDs so they are non-negative integers. Points which do not belong to any of the clusters (noise) are assigned zeros. Since $< Z_i, Z_l >= \sum_j Z_i[j] \cdot Z_l[j] = 0$ and based on the fact that when a sum of non-negative integers is zero, then all integers are zero, we obtain $Z_i[j] = 0$ or $Z_l[j] = 0$ for all $j = 1, 2, \ldots, n$.

For example, the clustering vectors
$Z_i = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 2\ 2\ 2\ 2\ 2]^T$
$Z_l = [1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T$
are mutually orthogonal and contain different clusters.

Martingale construction. Each density level $\epsilon_t, t = 1, 2, \ldots, T$ provides one clustering vector $C_{DBSCAN(\epsilon_t)}$ for all $t = 1, 2, \ldots, T$. As t increases, more clustering vectors are computed and we gain knowledge about the vector C.

In Eq. (1), we constructed a sequence of vectors $C^{(t)}, t = 1, 2, \ldots, T$, where each $C^{(t)}$ is orthogonal to all $C^{(1)}, C^{(2)}, \ldots, C^{(t-1)}$, from Lemma 1. The sum of all clustering vectors $C^{(1)} + C^{(2)} + \ldots + C^{(t-1)}$ has zeros as cluster IDs in the points which belong to the clusters of $C^{(t)}$. Therefore, $C^{(t)}$ is also orthogonal to $C^{(1)} + C^{(2)} + \ldots + C^{(t-1)}$. We use the orthogonality to show that the vector $C^{(1)} + C^{(2)} + \ldots + C^{(t)}$ is our "best prediction" for the final clustering vector C at stage t. The expected final clustering vector at stage t is:
$E[C|C_{DBSCAN(\epsilon_1)}, C_{DBSCAN(\epsilon_2)}, \ldots, C_{DBSCAN(\epsilon_t)}] = C^{(1)} + C^{(2)} + \ldots + C^{(t)}$.

Initially, the final clustering C vector is the zero vector O. Our knowledge about the final clustering vector up to stage t is restricted to $C^{(1)} + C^{(2)} + \ldots + C^{(t)}$ and finally, at stage $t = T$, we have gained all available knowledge about the final clustering vector C, i.e. $C = E[C|C_{DBSCAN(\epsilon_1)}, C_{DBSCAN(\epsilon_2)}, \ldots, C_{DBSCAN(\epsilon_T)}]$.

5 Experiments

5.1 Dataset description

The proposed methodology is evaluated on the 20newsgroups-mini dataset with 2000 articles, which is available on the UCI repository[3] and on 220 news articles, which are references to specific Wikipedia pages so as to ensure reliable ground-truth: the WikiRef220. We also use two subsets of WikiRef220, namely the WikiRef186 and the WikiRef150, in order to test DBSCAN-Martingale in four datasets of sizes 2000, 220, 150 and 115 documents respectively.

We selected these datasets because we focus on datasets with news clusters which are event-oriented, like "Paris Attacks November 2016" or they discuss about specific topics like "Barack Obama" (rather than "Politics" in general). We would tackle the news categorization problem as a supervised classification problem, because training sets are available, contrary to the news clustering problem where, for example, the topic "Malaysia Airlines Flight 370" had no training set before the 8^{th} of March 2014.

We assume that 2000 news articles is a reasonable upper bound for the number of recent news articles that can be considered for news clustering, in line with other datasets that were used to evaluate similar methods [25, 5]. In all datasets (Table 2) we extract uni-grams and bi-grams, assuming a Bag-of-Words representation of text. Before the extraction of uni-grams and bi-grams, we remove the SMART[4] stopwords list and we then stem the words using Porter's algorithm [24]. The uni-grams are filtered out if they occur less than 6 times and the bi-grams if they occur less than 20 times. The final bi-grams are normalized using tf-idf weighting and, in all datasets, the upper bound for the density level is taken $\epsilon_{max} = 3$. We generate a sample of $T = 5$ uniformly distributed numbers using R, for the initialization of Algorithm 1.

Table 1: DBSCAN results without LDA, for the 5 best values of ϵ and $minPts$. The DBSCAN-Martingale requires no tuning for determining ϵ and is able to extract all clusters for datasets (eg. WikiRef220) in which there is no unique density level to extract all clusters.

WikiRef150			**WikiRef186**			**WikiRef220**			**20news**		
ϵ	clusters	NMI	ϵ	clusters	NMI	ϵ	clusters	NMI	ϵ	clusters	NMI
0.8	3	0.3850	0.8	3	0.3662	0.8	3	0.3733	1.6	20	0.0818
0.9	4	0.4750	0.9	3	0.4636	0.9	3	0.4254	1.7	20	0.0818
1.0	3	0.4146	1.0	4	0.4904	1.0	4	0.5140	1.8	20	0.0818
1.1	3	0.4234	1.1	3	0.3959	1.1	3	0.4060	1.9	20	0.0818
1.2	1	0.1706	1.2	2	0.1976	1.2	3	0.4124	2.0	20	0.0818

[3] http://archive.ics.uci.edu/ml/datasets.html

[4] http://jmlr.csail.mit.edu/papers/volume5/lewis04a/a11-smart-stop-list/english.stop

Table 2: Estimated number of topics. The best values are marked in bold. The majority rule for 10 realizations of the DBSCAN-Martingale coincides with the ground truth number of topics.

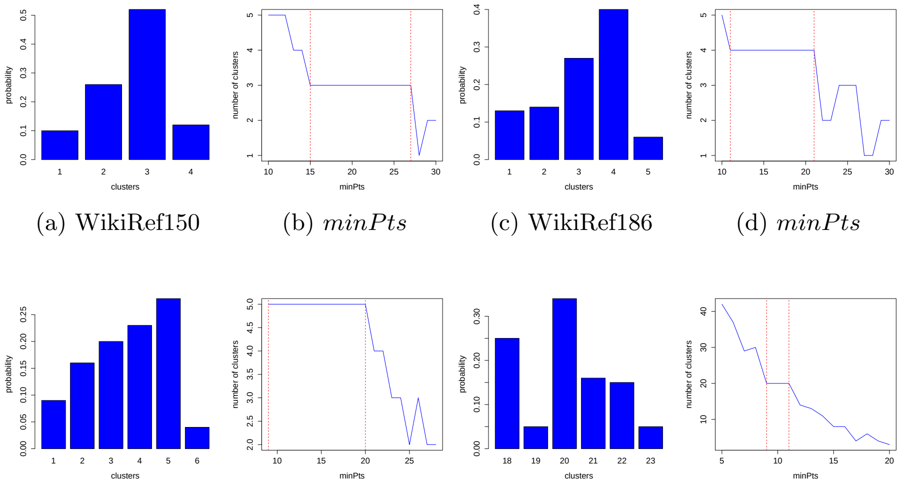
Index	**Ref**	**WikiRef150**	**WikiRef186**	**WikiRef220**	**20news**
CH	[6]	30	29	30	30
Duda	[9]	2	2	2	2
Pseudo t^2	[9]	2	2	2	2
C-index	[17]	27	2	2	2
Ptbiserial	[8]	11	7	6	30
DB	[8]	2	**4**	6	2
Frey	[13]	2	2	2	5
Hartingan	[15]	18	20	16	24
Ratkowsky	[26]	20	24	29	30
Ball	[3]	**3**	3	3	3
McClain	[22]	2	2	2	2
KL	[19]	14	15	17	15
Silhouette	[18]	30	**4**	4	2
Dunn	[10]	2	**4**	**5**	3
SDindex	[14]	4	**4**	6	3
SDbw	[14]	30	7	6	3
NbClust	[7]	2	2	6	2
DP-means	[20]	4	**4**	7	15
HDBSCAN-EOM	[5]	5	5	**5**	36
DBSCAN-Martingale		**3**	**4**	**5**	**20**

5.2 Evaluation

The evaluation of our method is done in two levels. Firstly, we test whether the output of the majority vote over 10 realizations of the DBSCAN-Martingale matches the ground-truth number of clusters. Secondly, we evaluate the overall hybrid news clustering framework, using the number of clusters from Table 2. The index "NbClust", which is computed using the NbClust[5] package, is the majority vote among the 24 indices: CH, Duda, Pseudo t^2, C-index, Beale, CCC, Ptbiserial, DB, Frey, Hartigan, Ratkowsky, Scott, Marriot, Ball, Trcovw, Tracew, Friedman, McClain, Rubin, KL, Silhouette, Dunn, SDindex, SDbw [7]. The Dindex and Hubert's Γ are graphical methods and they are not involved in the majority vote. The indices GAP, Gamma, Gplus and Tau are also not included in the majority vote, due to the high computational cost. The NbClust package requires as a parameter the maximum number of clusters to look for, which is set `max.nc = 30`. For the extraction of clusters from the HDBSCAN hierarchy, we adopt the EOM-optimization [5] and for the nonparametric Bayesian method DP-means, we extended the R-script which is available on GitHub[6].

[5] https://cran.r-project.org/web/packages/NbClust/index.html

[6] https://github.com/johnmyleswhite/bayesian_nonparametrics/tree/master/code/dp-means



(a) WikiRef150 (b) *minPts* (c) WikiRef186 (d) *minPts*

(e) WikiRef220 (f) *minPts* (g) 20news (h) *minPts*

Fig. 5: The number of clusters as generated by DBSCAN-Martingale

Evaluation of the number of clusters: We compare our DBSCAN-Martingale with baseline methods, listed in Table 2, which either estimate the number of clusters directly, or provide a clustering vector without any knowledge of the number of clusters. The Ball index is correct in the WikiRef150 dataset, HDBSCAN and Dunn is correct in the WikiRef220 dataset and the indices DB, Silhouette, Dunn, SDindex and DP-means are correct in the WikiRef186 datasets. However, in all datasets, the estimation given by the majority vote over 10 realizations of the DBSCAN-Martingale coincides with the ground truth number of clusters. In Figure 5, we present the estimation of the number of clusters for 100 realizations of the DBSCAN-Martingale, in order to show that after 10 runs of 10 realizations the output of Algorithm 1 remains the same. The parameter $minPts$ is taken equal to 10 for the 20news dataset and 20 for all other cases.

In all datasets examined, we observe that there are some samples of density levels $\epsilon_t, t = 1, 2, \ldots, T$ which do not provide the correct number of clusters (Figure 5). The "mis-clustered" samples are due to the randomness of the density levels ϵ_t, which are sampled from the uniform distribution. We expect that sampling from another distribution would result to less mis-clustered samples, but searching for the statistical distribution of ϵ_t is beyond the scope of this paper.

We also compared the DBSCAN-Martingale with several methods of Table 2, with respect to the mean processing time. All experiments were performed on an Intel Core i7-4790K CPU at 4.00GHz with 16GB RAM memory, using a single thread and the R statistical software. Given a corpus of 500 news articles, DBSCAN-Martingale run in 0.39 seconds, while the Duda, Pseudo t^2 and

Table 3: Normalized Mutual Information after LDA by k clusters, where k is estimated in Table 2. The standard deviation is provided for 10 runs and the highest values are marked in bold.

Index + LDA	**WikiRef150**	**WikiRef186**	**WikiRef220**	**20news**
CH	0.5537 (0.0111)	0.6080 (0.0169)	0.6513 (0.0126)	0.3073 (0.0113)
Duda	0.6842 (0.0400)	0.6469 (0.0271)	0.6381 (0.0429)	0.1554 (0.0067)
Pseudo t^2	0.6842 (0.0400)	0.6469 (0.0271)	0.6381 (0.0429)	0.1554 (0.0067)
C-index	0.5614 (0.0144)	0.6469 (0.0271)	0.6381 (0.0429)	0.1554 (0.0067)
Ptbiserial	0.6469 (0.0283)	0.6469 (0.0271)	0.8262 (0.0324)	0.3073 (0.0113)
DB	0.6842 (0.0400)	**0.7892 (0.0553)**	0.8262 (0.0324)	0.1554 (0.0067)
Frey	0.6842 (0.0400)	0.6469 (0.0271)	0.6381 (0.0429)	0.2460 (0.0198)
Hartingan	0.5887 (0.0157)	0.6513 (0.0184)	0.7156 (0.0237)	0.3126 (0.0098)
Ratkowsky	0.5866 (0.0123)	0.6201 (0.0188)	0.6570 (0.0107)	0.3073 (0.0113)
Ball	**0.7687 (0.0231)**	0.7655 (0.0227)	0.7601 (0.0282)	0.2101 (0.0192)
McClain	0.6842 (0.0400)	0.6469 (0.0271)	0.6381 (0.0429)	0.1554 (0.0067)
KL	0.6097 (0.0232)	0.6670 (0.0156)	0.7091 (0.0257)	0.3077 (0.0094)
Silhouette	0.5537 (0.0111)	**0.7892 (0.0553)**	0.8032 (0.0535)	0.1554 (0.0067)
Dunn	0.5805 (0.0240)	**0.7892 (0.0553)**	**0.8560 (0.0397)**	0.2101 (0.0192)
SDindex	0.7007 (0.0231)	**0.7892 (0.0553)**	0.8262 (0.0324)	0.2101 (0.0192)
SDbw	0.5537 (0.0111)	0.7668 (0.0351)	0.8262 (0.0324)	0.2101 (0.0192)
NbClust	0.6842 (0.0400)	0.6469 (0.0271)	0.8262 (0.0324)	0.1554 (0.0067)
DP-means	0.7007 (0.0231)	**0.7892 (0.0553)**	0.8278 (0.0341)	0.3077 (0.0094)
HDBSCAN-EOM	0.7145 (0.0290)	0.7630 (0.0530)	**0.8560 (0.0397)**	0.3106 (0.0134)
DBSCAN-Martingale	**0.7687 (0.0231)**	**0.7892 (0.0553)**	**0.8560 (0.0397)**	**0.3137 (0.0130)**

Dunn in 0.44 seconds, SDindex in 1.06 seconds, HDBSCAN in 1.23 seconds and Silhouette in 1.37 seconds.

Evaluation of news clustering: The evaluation measure is the popular Normalized Mutual Information (NMI), mainly used for the evaluation of clustering techniques, which allows us to compare results when the number of outputted clusters does not match the number of clusters in the ground truth [20]. For the output k of each method of Table 2, we show the average of 10 runs of LDA (and the corresponding standard deviation) in Table 3. For the WikiRef150 dataset, the combination of Ball index with LDA provides the highest NMI. For the WikiRef220 dataset, the combinations of HDBSCAN with LDA and Dunn index with LDA also provide the highest NMI. For the WikiRef186 dataset, the combinations of LDA with the indices DB, Silhouette, Dunn, SDindex and DP-means perform well. However, in all 4 datasets, our news clustering framework provides the highest NMI score and in the case of 20news dataset, the combination of DBSCAN-Martingale with LDA is the only method which provides the highest NMI score. Without using LDA, the best partition provided by DBSCAN has NMI less than 51.4 % in all WikiRef150, WikiRef186 and WikiRef220, as shown in Table 1. In contrast, we adopt the LDA method which achieves NMI scores up to 85.6 %. Density-based algorithms such as DBSCAN, HDBSCAN and DBSCAN-Martingale assigned too much noise in our datasets, a fact that affected the clustering performance, especially when compared to LDA in news clustering, thus we kept only the estimation $\hat{k}$.

6 Conclusion

We have presented a hybrid framework for news clustering, based on the DBSCAN-Martingale for the estimation of the number of news clusters, followed by the assignment of the news articles to topics using Latent Dirichlet Allocation. We extracted the word n-grams of a news articles collection and we estimated the number of clusters, using the DBSCAN-Martingale which is robust to noise. The extension of the DBSCAN algorithm, based on the martingale theory, allows for introducing a variable density level in the clustering algorithm. Our method outperforms several state-of-the-art methods on 4 corpora, in terms of the number of detected clusters, and the overall news clustering framework shows a good behavior of the proposed martingale approach, as evaluated by the Normalized Mutual Information. In the future, we plan to evaluate our framework using alternatice to LDA text clustering approaches, additional features and content, in order to present the multimodal and multilingual version of our framework.

Acknowledgements

This work was supported by the projects MULTISENSOR (FP7-610411) and KRISTINA (H2020-645012), funded by the European Commission.

References

1. Aggarwal, C. C., & Zhai, C. (2012). A survey of text clustering algorithms. In *Mining Text Data* (pp. 77-128). Springer US.
2. Ankerst, M., Breunig, M. M., Kriegel, H. P., & Sander, J. (1999, June). OPTICS: ordering points to identify the clustering structure. In *ACM Sigmod Record* (Vol. 28, No. 2, pp. 49-60). ACM.
3. Ball, G. H., & Hall, D. J. (1965). *ISODATA, a novel method of data analysis and pattern classification*. Stanford Research Institute (NTIS No. AD 699616).
4. Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *The Journal of machine Learning research*, 3, 993-1022.
5. Campello, R. J., Moulavi, D., & Sander, J. (2013). Density-based clustering based on hierarchical density estimates. In *Advances in Knowledge Discovery and Data Mining* (pp. 160-172). Springer Berlin Heidelberg.
6. Caliński, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1), 1-27.
7. Charrad, M., Ghazzali, N., Boiteau, V., & Niknafs, A. (2014). NbClust: an R package for determining the relevant number of clusters in a data set. *Journal of Statistical Software*, 61(6), 1-36.
8. Davies, D. L., & Bouldin, D. W. (1979). A cluster separation measure. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (2), 224-227.

9. Duda, R. O., & Hart, P. E. (1973). *Pattern classification and scene analysis* (Vol. 3). New York: Wiley.
10. Dunn, J. C. (1974). Well-separated clusters and optimal fuzzy partitions. *Journal of cybernetics*, 4(1), 95-104.
11. Doob, J. L. (1953). *Stochastic processes* (Vol. 101). Wiley: New York.
12. Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd* (Vol. 96, No. 34, pp. 226-231).
13. Frey, T., & Van Groenewoud, H. (1972). A cluster analysis of the D2 matrix of white spruce stands in Saskatchewan based on the maximum-minimum principle. *The Journal of Ecology*, 873-886.
14. Halkidi, M., Vazirgiannis, M., & Batistakis, Y. (2000). Quality scheme assessment in the clustering process. In *Principles of Data Mining and Knowledge Discovery* (pp. 265-276). Springer Berlin Heidelberg.
15. Hartigan, J. A. (1975). *Clustering algorithms*. New York: John Wiley & Sons.
16. Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of classification*, 2(1), 193-218.
17. Hubert, L. J., & Levin, J. R. (1976). A general statistical framework for assessing categorical clustering in free recall. *Psychological bulletin*, 83(6), 1072.
18. Kaufman, L., & Rousseeuw, P. J. (1990). *Finding groups in data. An introduction to cluster analysis*. Wiley Series in Probability and Mathematical Statistics. Applied Probability and Statistics, New York: Wiley, 1990, 1.
19. Krzanowski, W. J., & Lai, Y. T. (1988). A criterion for determining the number of groups in a data set using sum-of-squares clustering. *Biometrics*, 23-34.
20. Kulis, B., & Jordan, M. I. (2012). Revisiting k-means: New algorithms via Bayesian nonparametrics. *arXiv preprint*, arXiv:1111.0352.
21. Kumar, A., & Daumé, H. (2011). A co-training approach for multi-view spectral clustering. In *Proceedings of the 28th International Conference on Machine Learning* (ICML-11) (pp. 393-400).
22. McClain, J. O., & Rao, V. R. (1975). Clustisz: A program to test for the quality of clustering of a set of objects. *Journal of Marketing Research* (pre-1986), 12(000004), 456.
23. Milligan, G. W., & Cooper, M. C. (1985). An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2), 159-179.
24. Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3), 130-137.
25. Qian, M., & Zhai, C. (2014, November). Unsupervised feature selection for multi-view clustering on text-image web news data. In *Proceedings of the 23rd ACM international conference on conference on information & knowledge management* (pp. 1963-1966). ACM.
26. Ratkowsky, D. A., & Lance, G. N. (1978). A criterion for determining the number of groups in a classification. *Australian Computer Journal*, 10(3), 115-117.
27. Sander, J., Qin, X., Lu, Z., Niu, N., & Kovarsky, A. (2003). Automatic extraction of clusters from hierarchical clustering representations. In *Advances in knowledge discovery and data mining* (pp. 75-87). Springer Berlin Heidelberg.
28. Schneider, J., & Vlachos, M. (2013). Fast parameterless density-based clustering via random projections. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management* (pp. 861-866). ACM.
29. Teh, Y. W., Jordan, M. I., Beal, M. J., & Blei, D. M. (2006). Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476).