Information Retrieval Journal (2022) 25:239–268
https://doi.org/10.1007/s10791-021-09401-8

Measurement of clustering effectiveness for document collections

Meng Yuan[1] · **Justin Zobel**[1] · **Pauline Lin**[1]

Received: 7 June 2021 / Accepted: 16 December 2021 / Published online: 10 January 2022

Abstract

Clustering of the contents of a document corpus is used to create sub-corpora with the intention that they are expected to consist of documents that are related to each other. However, while clustering is used in a variety of ways in document applications such as information retrieval, and a range of methods have been applied to the task, there has been relatively little exploration of how well it works in practice. Indeed, given the high dimensionality of the data it is possible that clustering may not always produce meaningful outcomes. In this paper we use a well-known clustering method to explore a variety of techniques, existing and novel, to measure clustering effectiveness. Results with our new, extrinsic techniques based on relevance judgements or retrieved documents demonstrate that retrieval-based information can be used to assess the quality of clustering, and also show that clustering can succeed to some extent at gathering together similar material. Further, they show that intrinsic clustering techniques that have been shown to be informative in other domains do not work for information retrieval. Whether clustering is sufficiently effective to have a significant impact on practical retrieval is unclear, but as the results show our measurement techniques can effectively distinguish between clustering methods.

Keywords Evaluation of clustering · Effectiveness measures · Document clustering

1 Introduction

Clustering methods are used to partition sets of data items such that similar items will tend to be together. The use of clustering has a long history in computing, with the first applications in the 1950s (Bock 2007; MacQueen 1967; Forgy 1965), but the characteristics of the problem have remained fairly consistent. In particular, in the forms of

✉ Justin Zobel
jzobel@unimelb.edu.au

Meng Yuan
myuan3@student.unimelb.edu.au

Pauline Lin
pauline.lin@unimelb.edu.au

[1] School of Computing and Information Systems, University of Melbourne, Parkville, Australia

clustering of interest in this paper the resulting sets of items (or *clusters*) are disjoint, and the items are described by high-dimensional vectors.

The best-known clustering methods were developed for a range of applications in different fields but have been applied in information retrieval (IR) over many decades. Early in the history of IR, in the context of less powerful computers with limited online storage capacity, some researchers argued that clusters could form the basis of search methods (Voorhees 1986). This work followed principles encapsulated in the cluster hypothesis, namely that 'closely associated documents tend to be relevant to the same requests' (Jardine and van Rijsbergen 1971; van Rijsbergen 1979).

While these approaches were not adopted in practical systems, use of clustering in IR has continued. Note that some work in the field considers clustering of a corpus, while other work considers clustering of search results; our interest in this paper is in the former, and in particular where the number of items is large, as a means of pre-processing a corpus prior to retrieval. The clustering methods used for collections are primarily based on k-means (Liu and Croft 2004; Kulkarni and Callan 2010; Xu and Croft 1999; Broder et al. 2014; Cleuziou 2008) but use is also made of *hierarchical* clustering (Liu and Croft 2004; Kulkarni et al. 2012; Pfeifer and Leidner 2019; Jardine and van Rijsbergen 1971; Voorhees 1986).

In this paper, we explore how the effectiveness of clustering of text collections might be measured. Despite the uses made of clustering, to our knowledge there has been no previous examination of extrinsic measures of clustering quality in the context of IR, other than methods based on high-level manual topic labels, whose relationship to the needs of querying are arguable. Many measures have been proposed for clustering in general (Arbelaitz et al. 2013), but whether they are successful for text documents has not been tested, while there have been just a couple of methods for smaller collections or that rely on manual labelling (De Vries et al. 2012; Fuhr et al. 2012).

Key prior methods for measurement of clustering in general are intrinsic, that is, they rely on information within the clustered items. They include Dunn's index (Dunn 1973), the Davies–Bouldin score (Davies and Bouldin 1979), and the Silhouette measure (Rousseeuw 1987). These contrast within-cluster and between-cluster distances, between centroids and between items in the clusters. We also propose and make use of a new, simple intrinsic technique, *stability*; if clustering is genuinely reflective of semantic properties then independent clusterings—via different random seeds, or different methods—would be expected to be reasonably similar. All of these concern the internal structure of clusters, on the assumption that they will tend to be cohesive and have clear boundaries (Abraham et al. 2006).

Our first main contribution is to examine use of extrinsic techniques that are pertinent to IR: how the relevant documents for a query are distributed across clusters, and likewise the distribution across clusters of the documents retrieved in response to a query by a collection of systems. These measures are similar to those used as a benchmark for ranking of subcollections in distributed retrieval (Callan et al. 1995), but the purpose is different, namely to contrast the distribution of documents rather than the distribution of the clusters. A key additional factor is that we measure gain by comparison with an expected distribution given by random partitionings, as clustering into a small number of subcollections can give some degree of success by chance.

Stated concisely, then, our hypothesis is that *relevance judgements from ad-hoc retrieval evaluation can be used as the basis of extrinsic measures of clustering quality*. We do not claim that these measures are startling or profound, but they are the first proposed basis for extrinsic assessment of the quality of clustering of large and unlabelled document collections.

In our experiments, we use several existing and new measures to examine clusterings. By deliberately degrading the clustering algorithm we create a tool for seeing whether the loss of performance is reflected in the measures. The results show that two well-regarded intrinsic measures for clustering in general do not yield meaningful results for IR; the scores they report depend far more on the fidelity of the clustering to the item representation than they do to the quality of the clustering. Pleasingly, however, the results show that our proposed extrinsic measures do reflect the cluster quality. The richer the representation used to inform the clustering method, the better the score of the final result.

A secondary contribution of our work is examination of the extent to which standard clustering is indeed effective for IR tasks. The results show that clustering is somewhat effective: compared to the baseline or *natural* floor estimated by a random partitioning, there can be a significant reduction in the number of clusters that need to be inspected to access documents that are pertinent (relevant or retrieved) to a query. The extent of the reduction is dependent on collection- and task-specific parameters, but for some settings it is significant; in contrast to a plausible worst-case hypothesis, clustering does produce meaningful results.

2 Background on Clustering

Clustering has been used in many tasks in computer science and other disciplines; early studies involving clustering were mainly in bioinformatics and astronomy (Johnson 1967; Everitt et al. 2009). Three computing research fields where clustering is widely used are computer vision, for border and object detection tasks (Ester et al. 1996); machine learning, as an unsupervised approach to classification (Erman et al. 2006; Evans et al. 2011); and natural language processing, where for example it can be used to solve lexical ambiguity (Schütze 1992). Such work has been a driver of innovation in clustering.

There has been interest in clustering in IR for a range of applications. Early investigation of clustering in IR was encapsulated by the significance accorded to the cluster hypothesis, which was a central argument in a highly influential textbook (van Rijsbergen 1979).

However, there are counter-arguments to the cluster hypothesis. First, it does not necessarily follow that co-membership in a cluster means that the items are more similar than they are to items in general. As dimensionality increases proximity can become increasingly uninformative: an item can readily be similar to one neighbour in one set of dimensions and similar to another on other dimensions. That is, two items in separate clusters might each have good similarity to their respective centroids, but nonetheless be more similar to each other than they are to other items elsewhere in their clusters or indeed than they are to the centroids themselves; cluster boundaries are of necessity binary, but they may be arbitrary divisions through areas of ambiguity.

That is, it is possible that clustering of text is essentially spurious. In a space where the number of dimensions is so high that all objects might be at near-identical distances from each other, the outcome of clustering could primarily be dependent on marginal factors in weighting formulae and so on, and would not represent a semantically meaningful result. Our experiments explore this possibility.

Second, the proposal that the cluster hypothesis can form the basis of retrieval tends to imply that search topics also fall into a relatively small number of groups. If there are many more groupings of search topics than there are of documents, then only some search topics—perhaps only an insignificant fraction—will be well supported by any given clustering. (Our experiments illustrate that there is some level of misalignment between clusters and queries, as in every query we examined the pertinent documents are spread across multiple clusters.) However, the cluster hypothesis does offer an aim for clustering: to colocate items that will be retrieved together and are of similar topic. Our discussion of measures of clustering is based on this aim.

The concept of using clustering as the sole basis of search has not been pursued in practice. However, it is cognate to the problem of distributed retrieval, in which search is across a collection of disjoint corpora (Callan et al. 1995). It is also related to the concept of shards, which in some works are formed by clustering (Xu and Croft 1999; Kulkarni et al. 2012).

Clustering continues to be applied in information retrieval and for Web information analysis. An example is hot-event identification in social media (Becker et al. 2010; Abdelhaq et al. 2013; Li et al. 2012). This work aims at identifying the popular topics and shared interests over a platform by clustering texts and related contextual features. Li et al. propose a method to identify hot events on Twitter by applying Jarvis-Patrick clustering on tweet segments related to events; Abdelhaq et al. use keywords clustering followed by a scoring process; while Becker et al. create a feature space based on elements such as tags and location to learn a similarity metric for document clustering. In contrast to the early literature on clustering for IR, these methods are not limited to document retrieval.

Another application of clustering in IR is public opinion detection (Pal and Counts 2011; Yang and Miao 2018). Pal and Counts apply probabilistic clustering over features representing authors extracted from both nodal and topical metrics, and retrieve a list of authors who are active in a given topic; Yang and Miao use expectation maximisation to cluster synonyms for the same feature of a product in users' reviews and retrieve the general opinion on different facets of a product. Spam detection likewise makes use of clustering in both content-based and link-based approaches (Spirin and Han 2012).

2.1 Clustering Methods

A wide range of methods have been proposed for clustering of sets of data with high dimensionality, but only one is practical for large document corpora, k-means clustering, while hierarchical clustering can be used on smaller collections such as the set of documents that are relevant to a query. We introduce these below, and then discuss clustering methods for information retrieval.

K-means clustering. The clustering method most often used in IR is k-means clustering (Jain 2010). In this iterative approach, parameter k is the number of clusters. The initial step is a random choice of k items as seeds of k clusters. All other items are then assigned to the cluster based on least distance to the seed. In subsequent iterations, a centroid is computed for each cluster to be used as the reference point, and all items are then assigned to a cluster as before. The process stops when the centroids become stable at successive iterations, or when an iteration limit is reached.

Clustering methods such as k-means that use holistic properties are sometimes called top-down clustering. Refinements to k-means include k-value optimisation and successive partitioning of clusters (Kummamuru et al. 2003; Lydia et al. 2018; Modha and Spangler 2003). K-value optimisation is essential to tasks such as document grouping and document retrieval, because it should reflect the topical structure of the collection and will influence perceptions of whether clusters reflect meaningful aggregations of material. There are many methods described in the literature for choosing an optimal value of k from the point of view of effectiveness (Larsen and Aone 1999). However, it is agreed that there is no generalised way to predict an optimal value for k prior to clustering, since the document collections are usually unlabelled.

In the IR uses of clustering of which we are aware, the value of k in general seems to be predetermined, without explicit discussion of optimisation or consideration of alternatives, other than as an informal note that the value can be altered. The value chosen is generally in the range 10 to 100.

Elbow method. In this paper, we use the Elbow method (Thorndike 1953) for optimisation of k. The aim of the Elbow method is to maximise k while keeping overfitting to a minimum. In the Elbow method, there needs to be a scoring function for evaluating the clustering result and a range of k values.

A variety of scoring functions are used in the Elbow method. We use within-cluster sum of squared-error, here denoted W, a measure of the diversity of a set of items. For any clustering, as k increases, the average number of documents within clusters decreases. Considering the extremes, when k reaches the size of the whole corpus, overfitting exists in that cluster sparsity is at a minimum—there is only one document per cluster. On the other hand, if k is set to 1, then cluster sparsity is at a maximum and the clustering is not meaningful. Ideally, as k increases W decreases. At a certain point, the rate of decrease should slow down because the improvement in W contributed by increasing the number of clusters reaches a limit.

This turning point of the curve is the ‘elbow’. The value of k at the elbow is chosen. For clustering a collection of items D, the clustering is denoted as C, the clusters are denoted as c_i, the centroid of c_i as $\mathbf{c_i}$, and the Euclidean distance function as E. The value W for cluster c_i is calculated as follows.

$$W_{c_i} = \sum_{\mathbf{d_j} \in c_i} E(\mathbf{d_j}, \mathbf{c_i})^2$$

The W of a clustering C with k clusters is then $W = \frac{1}{k} \sum_{c_i \in C} W_{c_i}$.

Hierarchical clustering. The alternative to top-down is bottom-up, or agglomerative, clustering; hierarchical clustering is arguably the best known in this family (Fung et al. 2003). The method does not assume a fixed number of clusters, but instead, initially each item is assumed to be in its own cluster; these are then progressively merged, forming a tree structure (Johnson 1967). At each step the two closest clusters are merged. The algorithm stops when the number of clusters is sufficiently small.

Hierarchical clustering has been proposed as a method for supporting browsing (Tombros et al. 2002), by locating documents that are related to a query and visualising the relationship between the documents retrieved and the query, providing an alternative to the usual mechanism of listing all relevant documents as a one-dimensional array (Abualigah 2019; Bharti and Singh 2015; Cutting et al. 1992; Shafiei et al. 2007). In effect this approach treats the query as a root of a taxonomy, where the leaves are groups of documents related to the query in different dimensions.

For a large collection in a high-dimensional space, hierarchical clustering is infeasibly expensive. Its sensitivity and lack of need for prior parameterisation make it an attractive choice for tasks such as clustering the results returned in response to a query, but for our task, where we anticipate at a minimum hundreds of thousands of documents, it cannot be readily used.

Clustering for IR. Much prior research on clustering for IR, even including recent work, only makes use of small sets of documents, without evidence that the methods could generate meaningful clusters from retrieval-scale corpora. Whether they can scale algorithmically is in some cases also open to question.

In early work, Jardine and van Rijsbergen (1971) discussed the feasibility of applying hierarchical clustering for retrieval on the Cranfield Aeronautics collection of 200 documents; more ambitiously, Voorhees (1986) studied hierarchical clustering using single linkage and complete linkage on a collection of 12,000 documents. More recent work such as that of Avrachenkov et al. (2008) uses clustering on collections of up to 12,300 documents. While this work is designed to explore the effectiveness of the clustering methods for retrieval, they do not examine the challenges for scaling up to larger collections. A theme in such work is that the effectiveness of clustering is measured by how well the system works as a retrieval engine. For example, Jardine and van Rijsbergen (1971) use hierarchical clustering on a collection with 200 documents and 42 queries. For each query, a cluster is selected according to the search strategy, and precision and recall are calculated based on the documents in the cluster.

We note these works to illustrate the point that early research on cluster-based retrieval used small collections and the evaluation of clustering is different to ours; we study the feasibility of clustering rather than of retrieval. However, our extrinsic measures, being based on information associated with retrieval, can be seen as an adaptation and extension of these approaches.

Work on Web documents tends to use much larger collections (Kulkarni and Callan 2010; Liu and Croft 2004; Leuski 2001). There are no proposed clustering algorithms in these papers, but the way they use clustering varies. Liu and Croft (2004) build language models for documents in clusters generated by k-means and retrieval involves ranking of clusters. Kulkarni and Callan (2010) use a similar k-means method to that of Liu and Croft , except that they reduce computational costs by only clustering a sample of documents and projecting the rest of the collection to the clusters. Leuski (2001) uses hierarchical clustering and partition based on a threshold of cluster distance instead of the number of clusters. Again, these works do not make use of any measures specific to clustering, but rely on traditional measures of retrieval systems such as precision and recall.

An exception is Fuhr et al. (2012), which uses both k-means and hierarchical (agglomerative) clustering. They report that the latter is preferred, but restrict their attention to collections of under 10,000 documents each; the question of scaling is not considered.

It is not evident from first principles that clustering techniques that are effective in general will be successful for documents. Clustering rests on assumptions of proportionality and information density. For document collections, neither of these two assumptions may be valid. First, for any collection of substantive documents, the documents vary in length and type of content; information density does not have a simple relationship with document length. Clustering of such a vectorized document collection does not guarantee that documents with similar semantic meanings are in the same cluster and it cannot be assumed that the similarity of clusters' structure is a reliable measure of the effectiveness of document clustering. Second, as discussed above documents that are in the same cluster may (in vector space) be further from each other than they are from documents in other clusters, and document clusters are not necessarily cleanly separated.

Clustering was tested as a technique to increase the effectiveness of the SMART retrieval system (Salton 1971), where it was used to maximise term matching between queries and documents. As discussed above, clustering was regarded as a core component of retrieval by van Rijsbergen (1979). However, other researchers later concluded that the performance of document clustering was unreliable and mostly query-dependent (an outcome that we, in effect, re-examine in this paper). As noted by Willett (1988), the original clustering procedures did not adapt well to document collections and sometimes even reduced the performance of document retrieval when using clustering.

In recent decades, when clustering is applied to document corpora, documents and centroids are usually represented by feature vectors in which the features are normalised with standard TF-IDF weightings. Distance can then be computed using the Cosine measure (Croft et al. 2015), which is appropriate to this task because, in contrast to most current query similarity schemes, the two items whose distance is being estimated are treated symmetrically.

Kummamuru et al. (2003) proposed 'fuzzy' co-clustering of a document collection and keywords from the collection. In this work, there are several clusters as usual, but a document may be assigned to multiple clusters with different priority. This is a transition from the traditional single cluster assumption to a 'combination of topics' assumption. How the effectiveness of such clustering might be measured is beyond the scope of this paper, but we note that assessment of it has challenges that are not present in standard, disjoint clustering. Likewise, in light of the kinds of document description enabled by topic modelling (Wei and Croft 2006; Blei 2012; Ramage et al. 2009), methods were proposed that combine traditional clustering with topic modelling to enable multiple cluster assignment for documents; see Pfeifer and Leidner (2019) for a discussion of single versus multiple class membership (Pfeifer and Leidner 2019). Another approach is to distinguish degree of membership in soft or fuzzy clustering (Dunn 1973; Bezdek et al. 1984) from multiple class membership.

In agreement with most of the authors of the above, it is our view that it is naïve to assume that documents can be easily assigned to a single cluster such that the cluster will contain documents of the same topic. Obviously, a document may contain paragraphs that belong to different topics, and it is plausible that humans will often disagree on what a document label from a fixed, small set should be. Also, while measurement of clustering can make use of human labels, it is not obvious from first principles that it is feasible to attach labels to documents that are likely to reflect their topic for retrieval purposes. Studies have shown that full text provides a much better basis for retrieval than does metadata (Hawking and Zobel 2007); it seems implausible that a simple one-dimensional labelling could reflect the richness of the ways in which documents are accessed.

Nonetheless, clustering continues to be used, and for that reason it is valuable to consider how its effectiveness might be evaluated.

Pre-processing The clustering methods used for IR are adaptations of those developed for other domains, with modifications such as use of specific weightings and, as we now discuss, different approaches to pre-processing.

A key cost factor in clustering is the dimensionality of the data, which also influences effectiveness, as the boundaries of clusters can in principle become less distinct in high-dimensional space. Dimensionality reduction can be valuable when some dimensions are highly correlated or very sparse, because in this context doing so will clarify collection structure (Weber et al. 1998); both correlation and sparsity are evident in document corpora.

Options for dimensionality reduction in IR include feature selection, feature pruning, and orthogonalisation methods such as use of word embeddings (Modha and Spangler 2003; Abualigah 2019; Liu et al. 2015). Researchers have explored dimensionality reduction in order to speed up the clustering process for large data collections (Blott and Weber 2008; Cai et al. 2010; Shafiei et al. 2007; Weber et al. 1998; Xu et al. 2003). The approaches to dimensionality reduction can be summarised into two classes: feature selection and feature extraction (Liu et al. 2005). Feature selection focuses on selecting from a range of existing features in a dataset, such as the words in a document collection. In contrast, feature extraction creates secondary features from original data and these features may not be interpretable by humans; examples include word2vec (Mikolov et al. 2013) and Hidden Markov Models (Panuccio et al. 2002).

In this work, both feature selection and extraction methods are used. For feature selection, we use a bag-of-words approach with only the commonest features retained, then assign either binary weights or weights based on a standard TF-IDF formulation. For feature extraction, we use word-embedding method doc2vec (Le and Mikolov 2014), which is an extension of word2vec that supports embedding of paragraphs. We also apply feature pruning to the chosen methods to control the number of features used in clustering.

3 Background on Measurement of Cluster Quality

Our focus in this paper is on measurement of the effectiveness of clustering for information retrieval. We now introduce existing methods for measurement of clustering in general and on cluster measurement for IR.

As noted by Tomasini et al. (2016), ‘there is no unifying protocol for clustering evaluation, so it is often unclear which quality index to use in which case’. Arbelaitz et al. (2013) compare 30 measures of clustering, some of which are close variants of each other. A similar study, on synthetic data and a smaller number of measures, was undertaken by Tomašev and Radovanović (2016). The experiments by Arbelaitz et al. include results on 20 real, labelled datasets of up to 166 features and up to 2310 items—much smaller in both dimensions than is the case for document collections, but in the absence of larger-scale evaluations this work provides the best independent reference of which we are aware. These are intrinsic measures, that is, they rely only on properties of the items being clustered and do not make use of human labelling; they examine how cohesive clusters are and how well they are separated. In contrast, extrinsic measures use labels to consider properties such as cluster homogeneity or completeness.

To summarize the properties of intrinsic measures, Ben-David and Ackerman proposed four axioms for clustering measurements: isomorphism invariance, scale invariance, consistency, and richness. Although these features are labelled as ‘axioms’, they note that the term is not being used formally, due to the incomplete definition of clustering. Here we consider only the families of intrinsic measures that Arbelaitz et al. (2013) found to be superior in their experiments; it is unrealistic to expect ground truth of this kind to be available for large document collections. Specifically, we consider the Davies–Bouldin (DB) index (Davies and Bouldin 1979) and the Silhouette index (Rousseeuw 1987); these are explained below.

Following Arbelaitz et al. (2013), we define a collection D as a set of N documents where each document d is denoted as a vector with F dimensions. A *clustering* in D is a set of k disjoint sets (*clusters*) that partitions D into k groups: $C = \{c_1, c_2, \dots, c_k\}$ where

$\cup_{c \in C} c = D$ and $c_i \bigcap c_j = \emptyset, \forall i \neq j, 1 \leq i,j \leq k$. The *centroid* of cluster c is the mean vector, $\mathbf{c} = \frac{1}{|c|} \sum_{d \in c} \mathbf{d}$. Here, $\mathbf{d}$ is a vector representation of document d, and similarly for clusters. We denote the Euclidean distance between centroids of cluster c_i and c_j as $E(\mathbf{c_i}, \mathbf{c_j})$, and likewise between documents.

Abraham et al. (2006) proposed measurement of document clustering by the cohesion and separation of a clustering result; although the measurement technique is not novel to this work, it notes the need to evaluate the measurement of clustering for IR. Their approach does not rely on labelling of documents. Instead, it measures the quality of clustering by its internal structure. This evaluation metric rests on two assumptions. First, clusters should have a similar internal pattern to each other. That is, clusters should have similar size (within an order of magnitude), being neither dominant nor miniscule. Second, clusters should be well separated.

These are general principles for measurement of cluster quality. In general, *cohesion* can be defined in a variety of ways, including compactness (or equivalently density), the mean or median of within-cluster pairwise distances, or the maximum intra-cluster distance (or equivalently cluster diameter). The contrasting value *separation* is the distance between cluster centres.

The Silhouette index was found to be effective by Arbelaitz et al. (2013), and is reported in our experiments. First, two values a_i and b_i are required, where $d_i \in$ cluster c. The value a_i is the average intra-cluster distance from document d_i and b_i is the smallest average distance to the documents in any other cluster c'.

$$a_i = \frac{1}{|c| - 1} \sum_{d_j \in c, j \neq i} E(\mathbf{d_i}, \mathbf{d_j})$$

$$b_i = \min_{c' \neq c} \frac{1}{|c'|} \sum_{d_j \in c'} E(\mathbf{d_i}, \mathbf{d_j})$$

The Silhouette value s_i for a document d_i can then be calculated as follows.

$$s_i = \begin{cases} 1 - a_i/b_i & \text{if } a_i \leq b_i \\ b_i/a_i - 1 & \text{if } a_i > b_i \end{cases}$$

Then the Silhouette index, sc, is the average of the Silhouette value for all documents.

$$sc = \frac{1}{N} \sum_{i=1}^{N} s_i$$

For the Silhouette index, a score close to 1 is a good result; negative scores occur when the clustering is highly disordered. A score close to 0 means that the measure has not found overall structure.

Arbelaitz et al. (2013) similarly found that the Davies–Bouldin (DB) index was strong. It is defined as a ratio between the cluster scatter and the cluster's separation; a lower value will mean that the clustering is better. Under a formulation based on Euclidean distance, S_i is a measure of scatter within the cluster; it is the distance between documents and the centroid of the cluster.

$$S_i = \frac{1}{|c_i|} \sum_{d_j \in c_i} E(\mathbf{d_j}, \mathbf{c_i})$$

Let $R_{i,j}$ be

$$R_{i,j} = \frac{S_i + S_j}{E(\mathbf{c_i}, \mathbf{c_j})}$$

This is a measure of the amount of scatter within two clusters normalised by the distance between their centroids, and thus is a measure of the extent to which the clusters are separated. Then the Davies–Bouldin index is

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{i \neq j} R_{i,j}$$

Smaller values for the DB index indicate better clustering quality. We report results with both the DB index and the Silhouette index in our experiments.

As discussed by Abualigah (2019), some measures lack robustness for clusters of varying density and sizes. It is more robust to use external measurements to evaluate the correctness of clustering results based on ground truth, but as noted above the ground truth is not always available for document collections and labelling is subjective.

Another approach to examination of cluster quality is to use *stability*, that is, the extent to which the same clusters emerge regardless of method (Hennig 2007). In this work a variety of perspectives on stability are suggested, including that subsets of the collection should—if the clusters are meaningful—yield the same cluster structure as the full collection. However, the paper notes that stability by itself does not imply that the clusters are meaningful. The methods proposed by Hennig are not scaleable to our data; we explore a simpler approach to stability, as discussed later.

To our knowledge there has been only limited use of these measures for clustering effectiveness in IR or on text in general. Song and Park (2006) report use of the DB index as an optimisation target on a collection of 100 documents, but investigated how quickly the target was reached rather than how meaningful the clusters were.

Measurement of cluster quality on text (though not necessarily for IR) can be based on how well clusters match extrinsic, exhaustively curated high-level topic labels on the documents. As discussed earlier, we regard such labels as of questionable value; manual labelling of documents is not necessarily reflective of relevant to search (Hawking and Zobel 2007). However, it is an application of text clustering, as illustrated for example by Ingaramo et al. (2008) and Ingarmo et al. (2009), who seek to validate the reliability of several measures including the DB and Silhouette indexes on labelled text. On the assumption that the F-measure of the labels is a robust extrinsic assessment, they find a weak correlation, with wide spread, with these measures. In a similar vein, Zhang et al. (2015) use an intrinsic measure of clustering to generate labels for new text, again finding weak correlations. The implications of this work for IR are unclear.

Similar work that is more directly related to ours is reported by De Vries et al. (2012). The focus of their work is to show how divergence from a random baseline (which we call a natural floor or ceiling) on labelled text data can be assessed via labels; in this case as in the others discussed above these labels are topics, but here the labels are somewhat richer as multiple labels are allowed on each document. However, also as in other work reviewed above, the task for which the clusters are to be used in unspecified.

A richer investigation of measures of clustering for IR was undertaken by Fuhr et al. (2012), who proposed approaches based on queries and corresponding sets of documents, and reported results both for instantiations of these measures and for the intrinsic measures reported above. To our knowledge this is the only prior work comparing measures for IR. They note the possibility of using a query log and relevance judgements (amongst several other options), foreshadowing our approach but without consideration of how these approaches might differ in terms of validity.

Fuhr et al.'s experiments report what is in effect an intrinsic measure: the sets of documents are those that match 'keyphrases' extracted from the collection (which they call queries although they are not queries in the usual sense). This is equivalent to using one document representation for clustering and another for measuring the clustering. As only a single representation is used, and thus a single result for each measure on each collection, the results do not indicate whether the measures are robust; Fuhr et al. report that the Dunn index works well, for example, but this conclusion is based on high-level topic labels, in the same way as the work above, and thus has the same potential drawbacks. Another issue in this work is that the proposed measure requires a calculation that is quadratic in the size of the collection.

4 Clustering Measurement for IR

We now present our proposals for measurement of clustering quality in IR. In some applications of clustering, such as face recognition and genome sequencing, the quality of clustering can be validated by direct evidence because the datasets are quantitative and interpretable. However, document clustering in general does not have a direct measurement, and as discussed above labelling is neither feasible nor, necessarily, sufficiently meaningful—a cluster could easily be semantically plausible to a user but inconsistent with a fixed, 'universal' labelling. We therefore propose a collection of approaches to measurement, to offer a multi-faceted evaluation of clustering effectiveness.

After discussing cohesion and separation, we propose a new general measure of clustering behaviour, *stability*. We then describe our main contributions, measures catering to IR, based on relevant and retrieved documents. While such approaches have previously been speculatively noted as an option, to our knowledge, this is the first practical exploration and the first on a large collection.

4.1 Cohesion and Separation

Cohesion and separation were discussed in the previous section; the DB and Silhouette indexes are based on these properties. These are measures of intra-cluster and inter-cluster similarity. For clustering of documents with k-means, an obvious confound in using these measures is that separation of intra-cluster and inter-cluster similarity is the optimisation target for the clustering method, and a 'good' result could mean only that the algorithm had converged. Since k-means clustering depends on minimisation of Euclidean distances around centroids, if it is working it is expected to lead to clusters of (broadly) spherical shape and of similar size.

A degree of independence can be obtained by relying on different vectorisations of the data being clustered; for example, the distances used for measurement of cohesion and separation could rely on richer document vectorisations, while, for in-practice feasibility, the clustering might make use of some form of reduced dimensionality. Clearly, however, there will be correlation that undermines their value as measures.

They are not useless, however. It is possible that as collection size increases the quality of clustering will degrade, as the space becomes more dense; such degradation should be visible in measurements of cohesion and separation, or in convergence between measurement on a clustering and measurement on a random partition of the data, as discussed further in Section 4.5. Also, we have observed that the high dimensionality of clustering means that it is plausible that the results are essentially meaningless. If clustering acts as a completely random process, there should be no difference between the distribution of intra-distance and inter-distance values for any clustering configurations.

4.2 Stability

If a document collection consisted of k discrete, well-distinguished groups of thematically consistent documents, then it seems plausible that each run of a suitable and correctly parameterised algorithm would produce near-identical results. If instead the collection was homogeneous, each run—assuming that the clustering algorithm is not completely deterministic—would produce a completely different outcome; and likewise a poor algorithm would produce different outcomes. It is this variation in outcome that we seek to measure as *stability*.

Stability, an intrinsic measure (Hennig 2007), has not to our knowledge previously been proposed as a mechanism for assessing the reliability or value of clustering of documents. We propose a simple measure of stability as an further perspective on clustering of document collections, but note that Hennig and others explore richer definitions.

Considering a clustering as a set of subsets, and assuming the number of clusters k to be the same in each case, this is a measure of overlap between the subsets. There are several different approaches to such a calculation, with different justifications. We measure stability S as follows:

1. The two clusterings are $C_1 = c^1_1, \dots, c^1_k$ and $C_2 = c^2_1, \dots, c^2_k$.
2. Calculate the set of overlaps $o_{i,j} = \frac{|c^1_i \cap c^2_j|}{|c^1_i \cup c^2_j|}$, over all $1 \le i,j \le k$.
3. Repeating k times, choose the largest value of $o_{i,j}$ such that neither i nor j has previously been selected; the average S of these k values is the stability.

The k repetitions are required here to ensure that all clusters are considered.

This measure aligns the clusters according to their overlap, and has a high score (in the range 0.0–1.0) when many of the clusters are similar. It is a micro-average, though, with the limitation that small similar clusters can conceal poor overlap in large clusters. An alternative is a macro-average, where the measure is $\sum |c^1_i \cap c^2_j|/N$, the total number of documents in aligned clusters divided by the total in the collection; this can show good overall overlap but a single large, well-overlapped cluster can conceal poor performance in the smaller clusters.

Note that k-means clustering is sensitive to outliers and results in local, not global, convergence. This means that in principle it can be unstable. Whether this is the case in practice is assessed in our experiments.

4.3 Relevant Documents

The set of documents that have been judged relevant to a query are a form of extrinsic labelling. In a broad sense they constitute a subset of documents that are on a related topic. Critically, for our purposes, the topic is not derived from the collection but is conceived externally by a user, and judgements are made against those external criteria.

Given that the purpose of clustering in IR is to guide retrieval, a clustering where the relevant documents for queries were indeed together would be a successful and desirable outcome; that is, the extent to which the known relevant documents are gathered together is a strong indicator of the quality of the clustering.

A potential shortcoming is that the labelling is highly incomplete, with, in typical experiments, only a tiny fraction of the collection assessed for relevance. However, the judgements can be used as a sampling of the clusters; if the samples are sufficiently large, such as the hundred or so positive judgements per query across 50 or more queries observed in typical TREC experiments, then they can be reliably taken as indicative of cluster quality. Moreover, a document can be relevant to multiple queries, thus avoiding the naïvety of a simplistic labelling, and with a large set of queries a plural view of the clustering is provided.

A simple relevance-based measure would be to count the number of clusters that must be accessed to observe all of the known relevant documents, but such a measure would be sensitive to outliers—where just a few of many relevant documents had become widely spread.

For example, suppose that k-means had produced 20 clusters, and that for a particular query there were 50 relevant documents, 40 of them in 2 clusters and the remaining 10 across a further 8 clusters. (We have observed such distributions to commonly arise, as is illustrated later in our experimental results.) Measuring ‘the number of clusters that must be accessed to observe all of the known relevant documents’ would yield ‘10 of 20’, which suggests that the clustering was not particularly effective. However, a more nuanced reading of this result is that ‘2 of 20’ clusters provides 80% of the relevant documents, suggesting that the clustering is of good quality.

We therefore propose as a measure *clusters per query by relevance* to achieve a certain level, $p\%$, of coverage of the relevant documents, or

$$R_C@p = \frac{\text{minimum number of clusters to cover } p\% \text{ of the relevant documents}}{\text{total number of clusters}}$$

where p is a proportion such as 50% or 80%.

However, $R_C@p$ does not account for cluster size. Consider a clustering of 10,000 documents where the two largest clusters are 2500 documents each and the remaining 18 clusters are about 275 documents each. Continuing the example above, if the two clusters with 80% of the documents were the two largest, this would imply that half of the collection would have to be inspected to find the 40 relevant documents—better than random but not particularly impressive. On the other hand, if the two clusters with 80% of the documents were only 550 documents altogether, this would suggest that only a twentieth of the collection needed to be inspected to find the 40 relevant documents—an outcome that would reflect excellent clustering.

We therefore propose that the size of the clusters should also be considered, in the alternative measure *coverage per query by relevance*, or

$$R_V@p = \frac{\text{minimum total size of clusters to cover } p\% \text{ of the relevant documents}}{\text{total number of documents in those clusters}}$$

where p is again a proportion. We estimate R_V by sorting the clusters by decreasing density of relevant documents, then adding their sizes until the desired number of relevant documents has been observed. This is not the true minimum, because the correct computation is a bin-packing problem and thus NP-hard, but will be a close approximation when the numbers of clusters and relevant documents are small, as is the case in our experiments, and will be correct when the clusters are similarly sized.

As noted earlier, there is a correspondence between these measures and the approaches taken to measure techniques for collection selection (Xu and Croft 1999; Kulkarni et al. 2012). However, it is not a general assumption of collection selection that the subcollections should be gathered by topic, and the purpose of the measurement in our work is to compare divisions of material, not to compare retrieval methods.

Several works have argued against use of recall in IR measurements (Zobel 1998; Zobel et al. 2009). However, the point of clustering is, in some sense, to achieve recall after examination of a complete collection. The use here of a recall-like measure reflects the desire to show that material on a topic is indeed collected together, not to claim that all relevant documents have been found.

4.4 Retrieved Documents

An alternative to using documents that have been judged relevant is to use those that are retrieved by a system, or by a collection of systems, for a given query, up to some specified retrieval depth. In this approach no human judgement is required, but in our view the fact that the query is human-generated is critical; use of a query generated from the collection is effectively intrinsic, whereas an independent query reflects an extrinsic view of potential document content.

A measure based on retrieved material evaluates how much inspection of clusters is required to undertake the retrieval process. For instance, if the retrieved documents of a query are found in only two clusters, we can say that this clustering run describes the retrieval results of this query well; if all queries can be described by only a few clusters, the run has been effective at grouping documents. In contrast to intrinsic measures, use of retrieved documents from a system can verify whether a clustering run is useful independent of clustering methods or feature extraction approaches. A high score does not directly imply that clusters are truly thematic, but does imply that they have been successfully formed according to similarity to queries that reflect a human need.

By analogy with above, then, we propose *clusters per query by retrieval* (or fetching), or $F_C@p$, and *coverage per query by retrieval*, or $F_V@p$, as follows.

$$F_C@p = \frac{\text{minimum number of clusters to cover } p\% \text{ of the retrieved documents}}{\text{total number of clusters}}$$

$$F_V@p = \frac{\text{minimum total size of clusters to cover } p\% \text{ of the retrieved documents}}{\text{total number of documents in those clusters}}$$

The use of retrieved documents has similarities to the approach of Fuhr et al. (2012), but the calculation is much cheaper, as it is based on cluster cardinalities rather than properties of the individual documents.

A design decision in use of F_C and F_V is of what system, or systems, does the retrieving that produces the lists of documents. One approach is to have a single system, perhaps with some simple variations in similarity formulation, and run large numbers of queries (extracted from a query log, perhaps). However, this approach may tend to favour clustering methods that rely on the same forms of document representation as are used in the retrieval system, rather than reflect the semantics of the underlying data.

Our view is that diversity in the retrieval systems is required to give a reasonable level of confidence that the measures of clustering are reflective of the semantics of the data. In our experiments, computation of F_V and F_C used the runs from all systems to depth 10 to generate a per-query pool of retrieved documents. The reported values of F_V and F_C are thus averages over the same number of queries as R_V and R_C.

These experimental parameters could have been varied in a range of ways: a random subset of systems; the systems that were placed in the top 50% or 10% by a measure such as average precision; greater depth in runs; taking each system-query pair separately, rather than pooling; and so on. With our initial choices, as the results show the rankings of clusterings were reasonably similar with both R and F. Our expectation is that reducing the number of systems will have only limited effect, given the high overlap between runs, but deepening the pools could be a confound as it would increase the difficulty of gathering a large fraction of the documents into a small number of clusters.

4.5 Range and Gain

A key question with all of these measures is of what constitutes 'good' and 'bad' clustering. That is, for example, it may not be obvious for a given $R_C@80$ score whether it represents a meaningful outcome. To help make results comparable, we make use of estimates of the *natural* floor or ceiling in which, for a given clustering into clusters of certain sizes, we calculate the likely behaviour if the assignment of documents to clusters was random while preserving the cluster sizes.

By identifying a plausible minimum (or maximum) performance in this way, we identify the true range of values from which a score is drawn. Thus, for example if there are 20 clusters and 10 relevant documents, a good clustering might put all the relevant documents in 2 clusters while a random partitioning would put them in about 8. The performance of the clustering would then be to have achieved a reduction from 8 to 2, not, as might naïvely be assumed, from 20 to 2. The natural range of scores is from 1 to 8, not 0 to 20. (A score of 0 is unachievable, because the documents need to be in at least one cluster.) A similar approach was pursued by De Vries et al. (2012), who describe the natural value as a baseline.

Use of such a natural floor allows examination of the improvement or *gain* observed by a clustering, relative to the random partitioning. For example, suppose we are considering $R_C@p$ at $p = 80\%$. Here, gain is defined by $g(\cdot) = 1 - \frac{a(\cdot)-1}{e(\cdot)-1}$. We wish to find $g(R_C@80)$, where $a(R_C@80)$ is the minimum number of clusters required to attain 80% retrieval and $e(R_C@80)$ is the natural ceiling, with the '-1' adjustment needed because at least one cluster must be involved, thus allowing gain to reach 1.0. Gain is assumed to be 0.0 if $a(\cdot) > e(\cdot)$ or if $e(\cdot) = 1$.

Note that it is uninformative to measure performance on random partitioning using the DB or Silhouette indexes. Even for a collection of items of unbalanced distribution, the observed value of the Silhouette index will tend towards zero as the collection grows (a_i and b_i should be increasingly similar), while for the DB index the distance $E(\mathbf{c_i}, \mathbf{c_j})$ between centroids will tend towards zero, so the value of the index will be unbounded. We thus do not report an equivalent to gain for these intrinsic measures.

Gain could plausibly used to tune k, in a similar way to the Elbow method. Considering the extremes, if there is only one cluster the gain is always 0. If each document is in a cluster by itself, likewise the gain is always 0. Gain (for a particular measure, of course) will be maximised at some intervening value of k. We leave such exploration for future work.

5 Experimental Design

In our experiments we apply clustering to multiple data sets, using a range of clustering techniques that are chosen to create results that are by construction of varying quality. The measures of clustering, if they are reliable, should then reflect this variation. We now introduce these elements of the experimental design.

Data sets. We use four data sets in our work.

DISKS45 TREC disks 4 and 5 (Voorhees and Harman 2005), comprising 556,077 documents. For this data, there are 150 topics (301–450), comprising three subsets of 50 queries each, for each of which there is over 100 runs from the systems that participated in the TREC events. Per query, there is an average of 93 documents judged relevant and pool sizes (to depth 10) average around 298.

Note that we developed our methods on a 1% random subset of DISKS45. In our view this does not undermine the independence of results that are reported on the full collection.

Most of our reported results are on DISKS45, for ease of comparison across different measurement methods, but results were consistent on all of the data sets.

SMALL45. A 10% subset of DISKS45 that is drawn at random. The same queries are used. This data set is used to explore how the measures are affected by collection scale.

WT2G. The TREC-8 Web track, comprising 247,491 documents and 50 queries, with 44 contributing systems.

WT2G-C. In our experiments with WT2G we found that the clustering always tended to produce a single large cluster, with over half of the documents. Many of the documents in the large cluster were extremely short; around 70,000 had fewer than 10 tokens, all in the largest cluster. When we re-clustered without these short documents, roughly the same clusters were formed, except for the large cluster which was somewhat reduced. We report results on both the original WT2G and the reduced version without the short documents, WT2G-C.

Clustering methods. In all of our experiments we use k-means clustering.[1] As discussed, other approaches are not always practical, but in any case our goal here is not to establish the best possible clustering method, but to explore how variation in how documents are

[1] Specifically, from the `sklearn` package in Python with initiation mode `k-means++` to speed up convergence.

clustered is susceptible to measurement. We set the maximum number of iterations to 300 as k-means clustering does not always converge.

Our approach is to first use the Elbow method, with polynomial regression with an order of 10,[2] to establish a reasonable choice of k, which is then fixed for the remainder of our experiments.

To vary the quality of clustering, we use three different vectorisations of the documents.

BINARY The m most common words in the collection are identified; the representation of a document is a vector with a 1 if the word is present or 0 if the word is absent. We use m of 50 and 500, with the former value in particular chosen to ensure that the clustering is likely to poor.

TFIDF The m most common words in the collection are identified, after removal of stopwords and words occurring in more than 5% of the collection; the representation of a document is a vector with a TF-IDF weight of each the word (0 if the word is absent). We use m of 500 and 8000. The formulation of TF-IDF is as in Zobel and Moffat (2006). TFIDF and BINARY are examples of bag-of-word vectorisations.

DOC2VEC We generate DOC2VEC vectors representing each document (Le and Mikolov 2014) using gensim,[3] of length 100 and 500. Our expectation is that DOC2VEC 500 should be similar or superior to TFIDF of the same length, as it is a richer representation.

We do not report results with multiple values of k, though we note that in clusterings of different kinds comparison across k is also of value. This is because we already have a rich landscape of parameters to explore. The decision to use 8000 as the upper limit for TFIDF and 500 for DOC2VEC was determined by consideration of processing costs, over the large numbers of iterations required to build these results; k-means grows in computational cost as dimensionality is increased.

An alternative to use of BINARY as a poor representation would be to cluster with a reliable representation and then degrade by randomly swapping a controlled percentage of documents between clusters. This would have the advantage of showing whether the measures could smoothly track cluster quality, but would not illustrate whether the measures could detect a genuinely poor approach; as this was our primary aim we chose to use BINARY, but if results had been unclear would have also explored random degradation.

In each experiment, we ran k-means clustering 10 times with different initial seeds, and report averages over the 10 runs. For each clustering, there is a set of cluster sizes; the random partitioning used to estimate the natural floors or ceilings were generated to have the same set of sizes.

Clustering measurements. For each clustering, we compute the DB and Silhouette indexes, and $R_C@p$, $R_V@p$, $F_C@p$, and $F_V@p$ for proportions p of 50%, 80%, and 100%. We report gain for each of these, which we regard as more informative than the absolute measurements as they account for the natural range of possible scores.

[2] The implementation of regression is from https://seaborn.pydata.org.

[3] https://radimrehurek.com/gensim/.

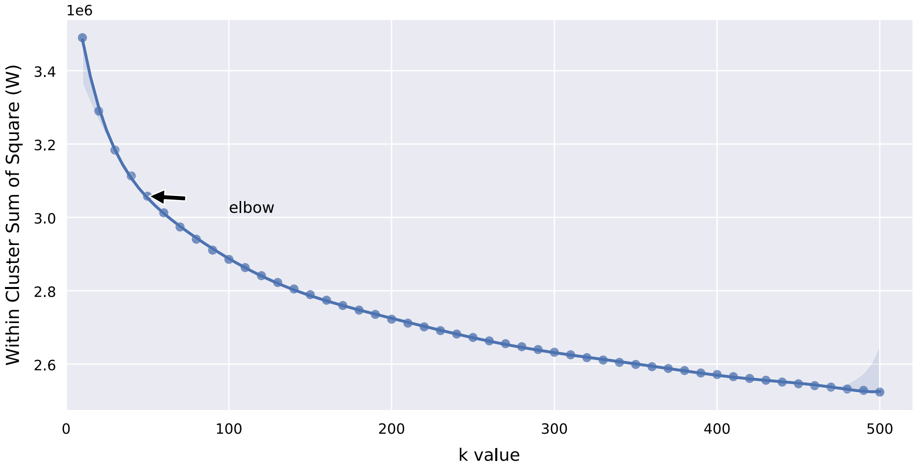


Fig. 1 Use of the Elbow method to find an appropriate k value for TFIDF 500 clustering on DISKS45. The figure shows average within-cluster sum of error W for a range of values of k. The 'elbow' of the curve is the best trade-off between optimisation over k and overfitting. The shadow around the curve represents a confidence interval. Since the values are discrete, we fit the data using regression with order of 10 to obtain the curve. Here, the best value of k is 50

6 Results

We now present our results. We first use the Elbow method and stability to explore the broad behaviour of k-means clustering on this data, then use visualisation to explore the behaviour of our proposed external measures. These measures are then compared to each other and to intrinsic measures to examine whether they are of value.

6.1 Elbow Method

To estimate how many innate clusters there are in the corpus, we apply the Elbow method for a range of k values from 10 to 500. Results are shown in Fig. 1 for TFIDF 500 on DISKS45; we observed similar patterns, and a similar turning point, for other document vectorisations. The shadow is the confidence interval of the regression. In the middle part of the curve, values are fitted more precisely than at the ends, hence the (small) area of shadow around the curve at $k = 10$ and $k = 500$. On close inspection, it can be seen that data points at these positions are off the curve.

These results show that a wide range of k values would be suitable; we chose $k = 50$ as it was the turning point, if only by a small margin. Many uses of clustering in previous IR work happen to use this value, though the basis of the choice is not always indicated. Our results do suggest, however, that the smaller values of k used in some work (such as 10) may not be appropriate.

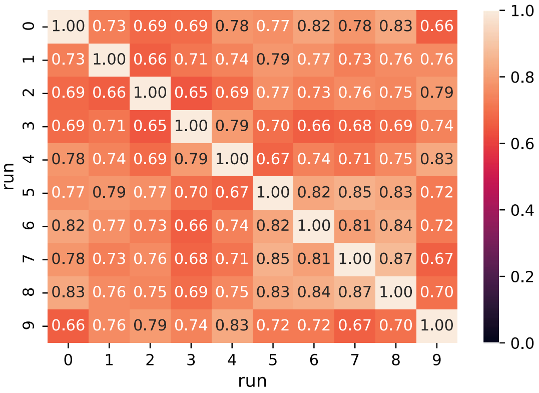


Fig. 2 Clustering stability of *k*-means on DISKS45 for TFIDF 8000 for *k* of 50. Ten runs of TFIDF 8000 clustering results are generated and their set overlap *S* is shown as a heat map. As shown in the figure, the clustering is stable, since the median set difference between two runs is over 70%

6.2 Stability

A fundamental question is whether clustering produces results that are meaningful at all, as discussed in the opening section of this paper. If *R* and *F* show significant gain (as discussed later), then arguably clustering is producing clusters that semantically distinct. If, however, every iteration produces unrelated results then the statistical validity of claims about gain would be questionable.

We use stability to explore whether clustering runs are reasonably consistent with each other. For DISKS45 and TFIDF 8000, we clustered 10 times (with *k* of 50, as in the remainder of our experiments).

The results are shown in Fig. 2 as a heatmap, where clusterings are compared on a pairwise basis between runs, as explained earlier. We used the micro-averaged form of *S* as the clusters were of broadly similar size, meaning that the potential confound we noted would not arise.

The results are clear. The median set overlap is over 70%, with a worst case of 66%; with 50 similarly sized clusters, a random overlap would be expected to be less than 5%. It can be concluded that the clustering is reasonably stable and is distant from random, while also noting that the clusters are not identical and thus the starting point does introduce some variation. We can therefore assume that observed measurements of cluster quality in our experiments arise from the clusters representing underlying (extrinsic) properties of the data—noting that this does not by itself imply that the clusters are semantically distinct in ways that are valuable for IR.

6.3 Intrinsic measurements

We now examine the behaviour of intrinsic measurements for different vectorisations. For the DB index, a lower value represents better clustering quality; for the Silhouette index, a value close to 1 represents better clustering quality, whereas a negative value means that documents are distributed rather than gathered.

Results are shown in Table 1, for each of the four collections and six vectorisations. As can be seen, the scores are chaotic. The DB index and Silhouette score are not consistent with each other, and implausibly the BINARY 50 clustering is often amongst the best. As can be seen, for the newswire collections the Silhouette score is always close to 0 and thus is

Table 1 Results for the six clusterings on the four collections as reported by intrinsic measures, the DB index and the Silhouette index

Clustering	DB index				Silhouette			
vectorisation	disks45	small45	wt2g	wt2g-c	disks45	small45	wt2g	wt2g-c
binary 50	3.64	3.68	3.30	4.10	−0.01	−0.01	**0.41**	0.05
binary 500	6.05	6.00	5.07	5.04	−0.08	−0.08	0.24	−0.01
tfidf 500	4.01	3.91	3.50	3.43	**0.07**	**0.07**	0.26	0.07
tfidf 8000	6.40	6.14	3.03	4.99	0.03	0.03	0.18	0.05
doc2vec 100	**3.42**	**2.88**	**2.58**	**2.59**	−0.08	−0.03	0.25	**0.09**
doc2vec 500	4.33	2.94	2.97	2.97	−0.10	−0.05	0.30	0.08

Bold results are the best score in that column

Table 2 DB index computed by each vectorisation for each clustering on disks45

Clustering	Evaluation vectorisation					
vectorisation	B50	B500	T500	T8000	D100	D500
binary 50	**3.64**	8.23	10.31	9.48	8.97	9.42
binary 500	7.17	**6.05**	9.01	9.03	8.88	9.37
tfidf 500	9.86	8.46	**4.01**	**4.92**	8.74	8.53
tfidf 8000	16.26	12.78	7.41	6.40	11.26	10.67
doc2vec 100	10.67	8.80	7.13	6.42	**3.42**	3.47
doc2vec 500	12.58	10.78	8.28	7.61	4.61	4.33

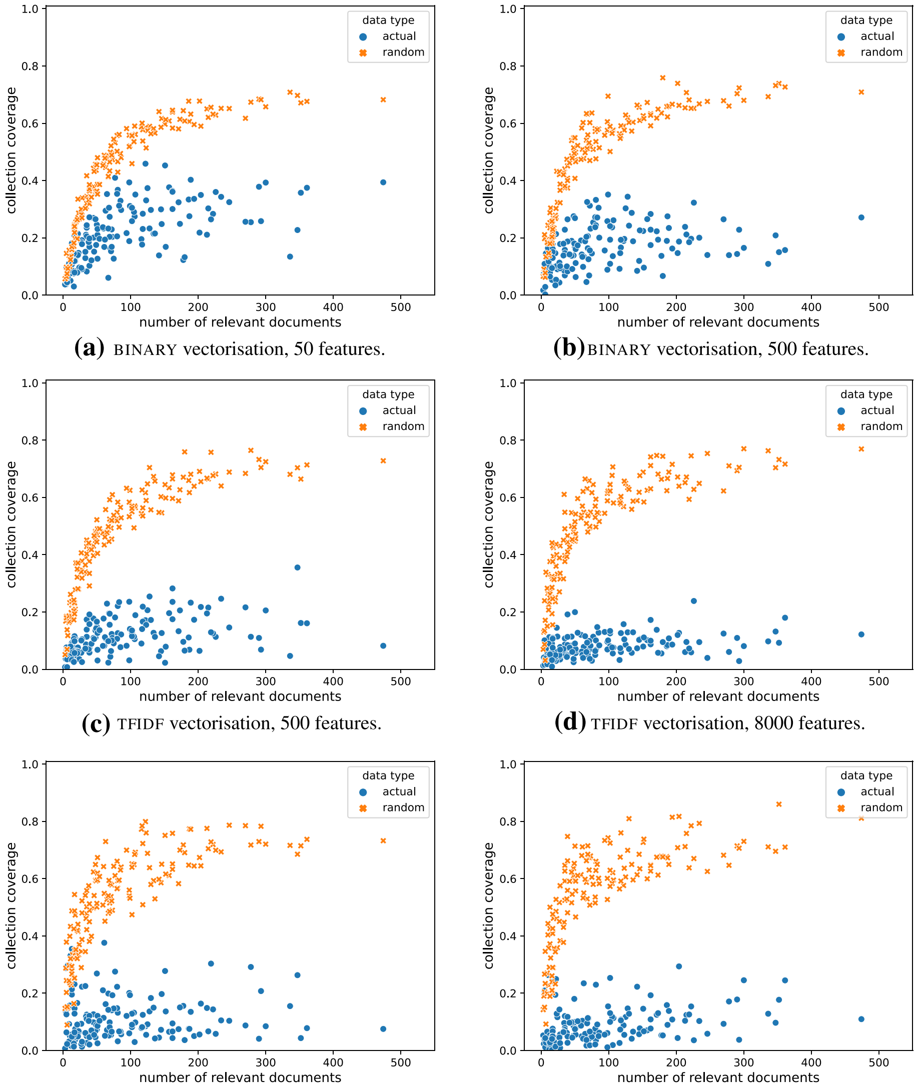
In the top row, B, T, and D are shorthand for binary, tfidf, and doc2vec respectively. Bold results are the best score for that measure in that column

not informative. Comparing wt2g and wt2g-c, the DB index is only somewhat altered by removal of the tiny documents, but the Silhouette score is drastically changed; these two indexes are not consistent with each other.

Note that the indexes yield very similar values for disks45 and small45. We discuss this further below in the context of results on the extrinsic measures.

In the measurements reported above, the vectorisation used to calculate the DB index was the same as that used to generate the clustering. In Table 2, we explore how the DB index behaves on disks45 when one vectorisation is used for clustering (one per row) and is then measured with the other vectorisations (one per column). For example, the value 10.31 at the intersection of row binary 50 and column T500 is the DB index of a clustering using binary vectorisation with the top 50 terms, as evaluated with calculations based on use of tfidfwith the top 500 terms.

As can be seen, the clustering vectorisation is always the best or second-best with respect to the evaluation vectorisation—thus demonstrating that this intrinsic measure is giving no indication of whether documents have in any external sense been usefully gathered together. Indeed, these results suggest that the second-best representation overall is binary 50 and the worst is tfidf 8000—showing that these measures are not semantically meaningful. That is, the intrinsic measures are only measuring the extent to which the clustering was performed with the same representation as that underpinning



(a) BINARY vectorisation, 50 features.

(b) BINARY vectorisation, 500 features.

(c) TFIDF vectorisation, 500 features.

(d) TFIDF vectorisation, 8000 features.

(e) DOC2VEC vectorisation, 100 features.

(f) DOC2VEC vectorisation, 500 features.

Fig. 3 Collection coverage R_V@80 for relevant documents on DISKS45. Each point represents a topic with the given number of documents relevant to a topic from TREC topics 301–450. A random document partition is generated with respect to the corresponding real clustering result to give the natural ceiling. For each topic, we determine the minimum total size of the clusters needed to cover 80% of the relevant documents appear and report collection coverage as fraction of the total corpus size. There is only a partial correspondence between number of relevant documents and coverage, but the richer vectorisations allow a much smaller coverage.

the measure; they are not reflecting fundamental properties of the data. We note that these results somewhat contradict those reported by Fuhr et al. (2012), again suggesting that topic labels do not necessarily align well with the needs of retrieval.

6.4 Collection Coverage

We now examine the behaviour of our external measures based on coverage per query topic. As shown in Sect. 6.1, clustering performance only slowly changes with k, but mathematically there is an elbow at 50 and thus we choose $k = 50$ for all clustering methods.

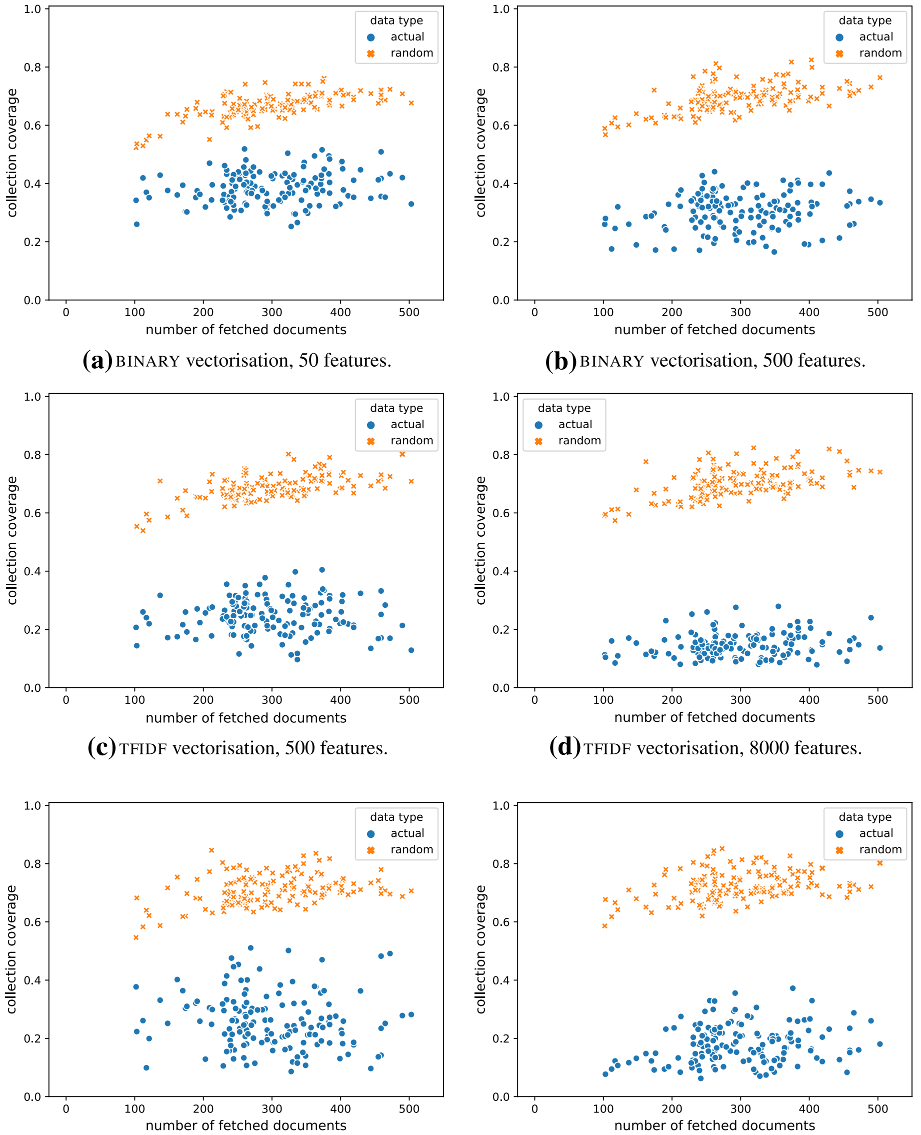
First, we compare the coverage per topic for clustering and its corresponding natural ceiling (as given by random partitioning) with respect to TREC relevance judgements. Results for R_V with $p = 80$ are shown in Fig. 3. In each of the six graphs, a different vectorisation is used to generate a clustering, giving 50 clusters of a range of sizes, which are then mimicked to give the random partitioning, that is, of the same distribution of sizes. Each query has a different number of documents that have been judged relevant, giving the spread along the x-axis.

The orange points show, per query, the value of R_V needed to achieve 80% coverage of the relevant documents on the random partitionings; as can be seen, unsurprisingly these are consistent across all the vectorisations, because the assignment of documents to clusters is random, but is not quite identical because there is some change in the distribution of the sizes of the clusters. (Note that the reported values are averages across 10 different partitionings and 10 different clusterings, with different random seeds.) More surprisingly, perhaps, even when there are over 100 relevant documents these values are much lower than 1.0—in this range only 50%–70% of the collection needs to be accessed to find 80% of the relevant documents even under a random partitioning. For smaller numbers of relevant documents, even with random partitioning only a small number of clusters is needed, that is, the natural ceiling is low.

The blue points show, per query, the corresponding values of R_V for the actual clusters (again, these are averages across 10 different clusterings). Lower values reflect better clustering and thus the results show that the relevant documents are being distributed into clusters in a way that is clearly biased—which is the goal of clustering. Comparing the left column with the right, the trend is that increasing the number of features reduces R_V, showing the value of a richer representation. Comparing the rows, the designed-to-be-poor BINARY vectorisation does indeed give worse performance than the others, confirming that R_V reflects cluster quality. The distribution of points makes comparison of TFIDF and DOC2VEC less clear, though overall TFIDF appears superior; we revisit this question below. As the plot for TFIDF 8000 shows, typically only around 10% of the collection needs to be accessed regardless of the number of relevant documents.

Corresponding results for retrieved documents, that is F_V, are shown in Fig. 4, again with $p = 80$. As noted, the top 10 documents are selected from each run to create a pool of retrieved documents. As the smallest pool is around 100 documents in Fig. 4, the appearance is somewhat different to that of Fig. 3, but in fact, account for the range, the patterns are of the same shape. Visually, while BINARY is weaker than the alternatives and TFIDF 8000 is the best performing, the trends here are less pronounced; however, they are clarified in the gain results reported below.

Continuing with DISKS45, the average gain of each vectorisation is shown in Table 3, for p of 50, 80, and 100. Interpreting these results, as an example a gain of 0.75 for R_V at $p = 80$ means that, compared to a random partitioning, on average only a quarter of the volume of documents needs to be accessed. As shown in Fig. 3, even a random partitioning often means that only 50%–70% of the volume of documents needs to be accessed even for large numbers of documents, so a gain of 0.75 implies that this volume is reduced to well below 20% of the collection.



(a) BINARY vectorisation, 50 features.

(b) BINARY vectorisation, 500 features.

(c) TFIDF vectorisation, 500 features.

(d) TFIDF vectorisation, 8000 features.

(e) DOC2VEC vectorisation, 100 features.

(f) DOC2VEC vectorisation, 500 features.

Fig. 4 Collection coverage F_V@80 for fetched documents on DISKS45. As for Fig. 3, random document sets are generated per clustering result, but the collection coverage is based on fetching retrieved documents in the top 10 positions in submitted runs, creating a pool. The smallest pool is around 100 items, hence the gap at the left of each plot. The BINARY vectorisations are the weakest.

Either TFIDF 8000 or DOC2VEC 500 has the best performance under each measure, with DOC2VEC slightly weaker, in a few cases being indistinguishable from DOC2VEC 100 and in other cases being weaker than TFIDF 500; when TFIDF is superior to DOC2VEC, the margin is usually substantial, which is not the case in reverse. These results also show that values for R and F are consistent with each other, in terms of ordering of representations.

Table 3 Gains using the four measures and six clusterings, for each of three proportions, for DISKS45

Clustering	Gain on measure					
vectorisation	R_V@50	R_V@80	R_V@100	R_C@50	R_C@80	R_C@100
BINARY 50	0.56	0.48	0.32	0.58	0.44	0.28
BINARY 500	0.69	0.64	0.50	0.60	0.50	0.37
TFIDF 500	0.77	0.74	0.56	0.79	0.66	0.46
TFIDF 8000	0.84	0.81	**0.65**	0.72	**0.72**	**0.56**
DOC2VEC 100	0.83	0.77	0.52	0.80	0.68	0.49
DOC2VEC 500	**0.86**	**0.83**	0.59	**0.83**	0.67	0.46
	F_V@50	F_V@80	F_V@100	F_C@50	F_C@80	F_C@100
BINARY 50	0.56	0.42	0.11	0.52	0.33	0.10
BINARY 500	0.71	0.56	0.21	0.49	0.32	0.17
TFIDF 500	0.78	0.64	0.24	0.72	0.48	0.21
TFIDF 8000	0.84	0.73	**0.31**	**0.79**	**0.56**	**0.27**
DOC2VEC 100	0.80	0.64	0.18	0.68	0.35	0.18
DOC2VEC 500	**0.87**	**0.75**	0.18	0.73	0.42	0.17

Bold results are the best score for that measure and proportion

Because the natural range is different for the different choices of p, what is not evident here is how small the fraction of the collection becomes for $p = 50$, but it is well below 10% for the better representations. In contrast, locating 100% of the pertinent documents means that all outliers must be found, leading to low gain in most cases. Our intuition was that $p = 100$ might be unreliable, because it is heavily dependent on outliers (hence our focus on $p = 80$ in the results reported above), but we reported it here for completeness. The results show that it is indeed less distinguishing than with the other choices of p.

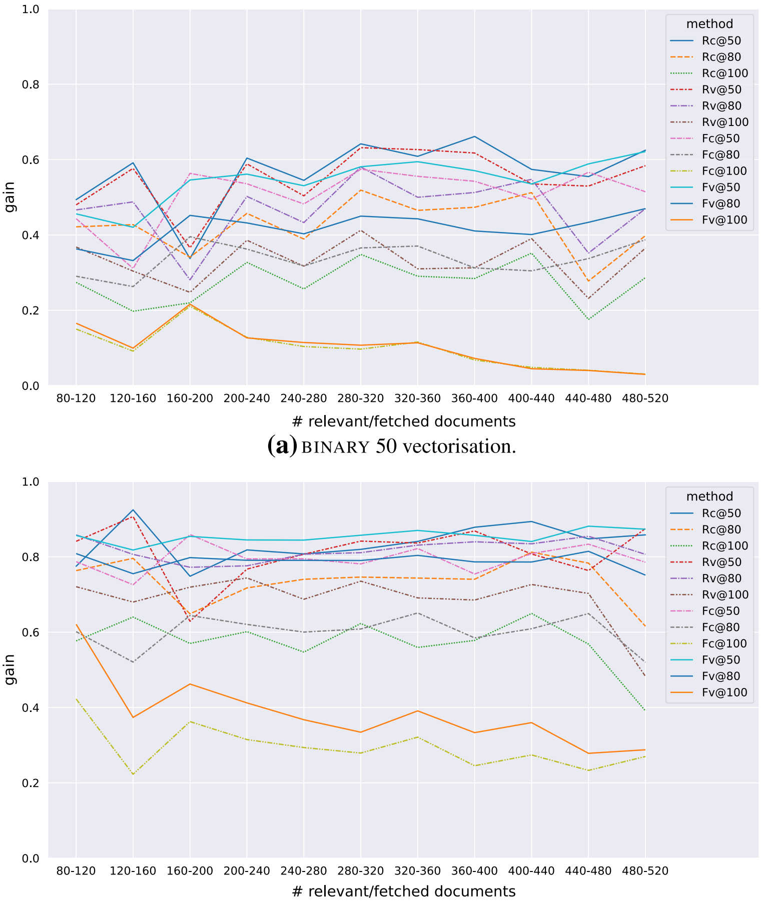
We complete our analysis of the extrinsic measures on DISKS45 with the results shown in Figure 5. These show gain on all measures at $p = 80$, binned by the number of pertinent documents per query, for BINARY 50 and TFIDF 8000—respectively our worst and best vectorisations as reported above. These results show the consistent increase in gain across all of our measures between the two vectorisations. With around 150 individual values in each of these graphs, in every instance TFIDF is superior to BINARY.

Finally, Table 4 shows gain values for $p = 80$ on all four collections. In these results TFIDF 8000 is consistently best or second-best, and TFIDF is always better than BINARY; while the DOC2VEC results are highly inconsistent.

We make several observations based on these results.

First, they show the value of gain as a measure, with consistent good performance by a particular representation. We hypothesised that either TFIDF 8000 or DOC2VEC 500 would yield the best clustering, and this has proven to be reflected in the measures.

Second, however, Table 4 does show that F_V and R_V, and F_C and R_C, are behaving somewhat differently from each other – as can be seen by comparing DISKS45 with SMALL45. The likely explanation is that the latter collection has only a tenth of the relevance judgements (influencing R), while the number retrieved (influencing F) stays the same. Taking the F values as more directly comparable than are the R values between the two collections, the results suggest that the quality of clustering degrades with collection size. If so, there are significant implications for prior work based on



(a) binary 50 vectorisation.

(b) tfidf 8000 vectorisation.

Fig. 5 Comparison of gain with all measurements on the binary 50 and tfidf 8000 vectorisations of disks45. The topics are partitioned into bins based on the number of relevant (respectively, retrieved) documents available. Note the dramatically better gain with the better vectorisation, and how the gain only drops slightly as the task becomes more challenging, that is, when there are more documents to fetch.

small collections, typically of only a few thousand documents, because these promising results may not scale to realistic collections. (Noting that many of these methods are too computationally costly for collections of the scale that we are considering.)

Third, the imbalance in the cluster sizes in wt2g and wt2g-c amplifies the difference between the ‘C’ and the ‘V’ measures; the contrasts in score are because the methods access many clusters but the are small. Nonetheless, both kinds of measure remain informative. The fact that many of the clusters few contentful documents allows the best representations to achieve very high gain, and shows that document quality as much as document topic is likely to be influential in the success of tfidf 8000.

Table 4 Measurements of gain for each of the four collections at $p = 80$

Clustering	disks45		small45		wt2g		wt2g-c	
vectorisation	R_V@80	R_C@80	R_V@80	R_C@80	R_V@80	R_C@80	R_V@80	R_C@80
binary 50	0.48	0.44	0.35	0.28	0.89	0.33	0.63	0.30
binary 500	0.64	0.50	0.43	0.30	0.90	0.31	0.82	0.31
tfidf 500	0.74	0.66	0.58	**0.49**	0.79	0.46	0.87	0.65
tfidf 8000	0.81	**0.72**	**0.65**	0.48	**0.99**	**0.67**	0.89	**0.75**
doc2vec 100	0.77	0.68	0.48	0.15	0.93	0.32	**0.89**	0.43
doc2vec 500	**0.83**	0.67	0.57	0.20	0.94	0.34	0.88	0.47
	disks45		small45		wt2g		wt2g-c	
	F_V@80	F_C@80	F_V@80	F_C@80	F_V@80	F_C@80	F_V@80	F_C@80
binary 50	0.42	0.33	0.58	0.55	0.80	0.24	0.63	0.17
binary 500	0.56	0.32	0.66	0.51	0.89	0.23	0.82	0.17
tfidf 500	0.64	0.48	0.77	0.71	0.76	0.45	0.80	0.56
tfidf 8000	0.73	**0.56**	**0.86**	**0.74**	**0.99**	**0.78**	0.89	**0.62**
doc2vec 100	0.64	0.35	0.48	0.17	0.97	0.32	**0.91**	0.30
doc2vec 500	**0.75**	0.42	0.65	0.27	0.96	0.33	0.90	0.32

Bold results are the best score for that measure and collection

7 Conclusions

We have shown that retrieval-based information can be used to measure the quality of document clustering. Documents judged as relevant to a set of queries provide a form of extrinsic evaluation that separates clustering based on simplistic document representations from clustering that makes use of richer information.

These results correlate closely with those based on documents retrieved by systems. Since document retrieval has similarities with the clustering mechanisms, there is potential for this to be a circular result (the measure and the clustering have properties in common). However, the close alignment with the relevance-based results, and the wide diversity of retrieval mechanisms in the systems that contributed to the test collections used, strongly suggests that the results are well founded. This means that no human assessment is needed in measurement of cluster quality, but only access to retrieval systems and a collection of queries.

This is the first work of which we are aware to measure the quality of document clustering for large corpora. Most techniques for measurement of clustering are intrinsic, but our results with such techniques show that they are indeed circular, and do not correspond to the underlying quality of the clustering.

Happily, our results show that there is strong skew in the gathering of documents during clustering, thus not confirming the hypothesis that the volume of material and high dimensionality might lead to meaningless results. They also show that the clustering is similar between runs with different random seeds, suggesting that the clustering is grouping documents in a way that is consistent with their contents.

However, care does need to be taken with use of clustering. We used k-means and chose k based on the Elbow method, which is at least independent of the use of the clusters, but no principles for choice of k in IR have been articulated, and most previous work assumes *k* without exploration; further work could experiment with other values of *k* and with other clustering algorithms. Also, there is only a degree of alignment between querying and clustering – our results show that the documents that are pertinent to a query are almost always spread across multiple clusters – and it is possible that clustering quality degrades with scale. The value of clustering of collections for general retrieval tasks thus remains an open question.

Overall, though, in this first systematic work on measuring clustering of document collections we have shown that it is feasible and informative. While further work is needed to explore their limits and sensitivity, our results and measures already provide a practical foundation for robust use and measurement of clustering in information retrieval.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions.

References

Abdelhaq, H., Sengstock, C., & Gertz, M. (2013) Eventweet: online localized event detection from twitter. In *Proceedings of VLDB international conference on very large databases* (vol 6, pp. 1326–1329) https://doi.org/10.14778/2536274.2536307

Abraham, A., Das, S., & Konar, A. (2006) Document clustering using differential evolution. In *IEEE international conference on evolutionary computation* (pp. 1784–1791), https://doi.org/10.1109/CEC.2006.1688523

Abualigah, L. M. Q. (2019). *Feature selection and enhanced krill herd algorithm for text document clustering*. Springer.

Arbelaitz, O., Gurrutxaga, I., Muguerza, J., Pérez, J. M., & Perona, I. (2013). An extensive comparative study of cluster validity indices. *Pattern Recognition, 46*(1), 243–256. https://doi.org/10.1016/j.patcog.2012.07.021

Avrachenkov, K., Dobrynin, V., Nemirovsky, D., Pham, S.K., & Smirnova, E. (2008) Pagerank based clustering of hypertext document collections. In *Proceedings of ACM-SIGIR international conference on research and development in information retrieval* (pp. 873–874) https://doi.org/10.1145/1390334.1390549

Becker, H., Naaman, M., & Gravano, L. (2010) Learning similarity metrics for event identification in social media. In *Proceedings of ACM international conference on web search and data mining* (pp. 291–300) https://doi.org/10.1145/1718487.1718524

Ben-David, S., & Ackerman, M. (2008) Measures of clustering quality: a working set of axioms for clustering. In: *Advances in neural information processing systems* (vol 21, pp. 121–128)

Bezdek, J. C., Ehrlich, R., & Full, W. (1984). Fcm: the fuzzy c-means clustering algorithm. *Computers&amp; Geosciences, 10*(2), 191–203. https://doi.org/10.1016/0098-3004(84)90020-7

Bharti, K. K., & Singh, P. K. (2015). Hybrid dimension reduction by integrating feature selection with feature extraction method for text clustering. *Expert Systems with Applications, 42*(6), 3105–3114. https://doi.org/10.1016/j.eswa.2014.11.038

Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM, 55*(4), 77–84. https://doi.org/10.1145/2133806.2133826

Blott, S., & Weber, R. (2008) What's wrong with high-dimensional similarity search. In *Proceedings of VLDB international conference on very large databases*, Auckland, New Zealand, https://doi.org/10.14778/1453856.1453861

Bock, H. (2007) *Clustering methods: a history of k-Means algorithms* (pp. 161–172) Springer Berlin Heidelberg, Berlin, Heidelberg https://doi.org/10.1007/978-3-540-73560-1_15

Broder, A., Garcia-Pueyo, L., Josifovski, V., Vassilvitskii, S., & Venkatesan, S. (2014) Scalable k-means by ranked retrieval. In *Proceedings of ACM international conference on web search and data mining, association for computing machinery* (pp. 233–242) New York, NY, USA, WSDM '14 https://doi.org/10.1145/2556195.2556260

Cai, D., He, X., & Han, J. (2010). Locally consistent concept factorization for document clustering. *IEEE Transactions on Knowledge and Data Engineering, 23*(6), 902–913. https://doi.org/10.1109/TKDE.2010.165

Callan, J.P., Lu, Z., & Croft, W.B. (1995) Searching distributed collections with inference networks. In *Proceedings of ACM-SIGIR international conference on research and development in information retrieval, association for computing machinery* (pp. 21–28) New York, NY, USA, SIGIR '95 https://doi.org/10.1145/215206.215328

Cleuziou, G. (2008) An extended version of the k-means method for overlapping clustering. In *International conference on pattern recognition* (pp. 1–4) https://doi.org/10.1109/ICPR.2008.4761079

Croft, W.B., Metzler, D., & Strohman, T. (2015) *Search engines: information retrieval in practice*. Originally published by Pearson

Cutting, D.R., Karger, D.R., Pedersen, J.O., & Tukey, J.W. (1992) Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of ACM-SIGIR international conference on research and development in information retrieval, association for computing machinery* (pp. 318–329) New York, NY, USA, SIGIR '92, https://doi.org/10.1145/133160.133214

Davies, D. L., & Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-, 1*(2), 224–227. https://doi.org/10.1109/TPAMI.1979.4766909

De Vries, C.M., Geva, S., & Trotman, A. (2012) *Document clustering evaluation: divergence from a random baseline*. https://arxiv.org/abs/1208.5654

Dunn, J. C. (1973). A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *Journal of Cybernetics, 3*(3), 32–57. https://doi.org/10.1080/01969727308546046

Erman, J., Arlitt, M., & Mahanti, A. (2006) Traffic classification using clustering algorithms. In *Proceedings of SIGCOMM workshop on mining network data* (pp. 281–286) https://doi.org/10.1145/1162678.1162679

Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of international conference on knowledge discovery and data mining* (pp. 226–231) AAAI Press.

Evans, R., Pfahringer, B., & Holmes, G. (2011) Clustering for classification. In *2011 7th international conference on information technology in Asia* (pp. 1–8) https://doi.org/10.1109/CITA.2011.5998839

Everitt, B. S., Landau, S., & Leese, M. (2009). *Cluster analysis* (4th ed.). Wiley Publishing.

Forgy, E. W. (1965). Cluster analysis of multivariate data : efficiency versus interpretability of classifications. *Biometrics, 21*, 768–769.

Fuhr, N., Lechtenfeld, M., Stein, B., & Gollub, T. (2012). The optimum clustering framework: implementing the cluster hypothesis. *Information Retrieval, 15*(2), 93–115.

Fung, B.C., Wang, K., & Ester, M. (2003) Hierarchical document clustering using frequent itemsets. In *Proceedings of SIAM international conference on data mining* (pp. 59–70) SIAM https://doi.org/10.1137/1.9781611972733.6

Hawking, D., & Zobel, J. (2007). Does topic metadata help with web search? *Journal of the American Society for Information Science and Technology, 58*(5), 613–628. https://doi.org/10.1002/asi.20548

Hennig, C. (2007). Cluster-wise assessment of cluster stability. *Computational Statistics&amp; Data Analysis, 52*(1), 258–271.

Ingaramo, D., Rosso, P., & Errecalde, M. (2008) In: *CICLing international conference on computational linguistics and intelligent text processing* (pp. 555–567), lNCS 4919

Ingarmo, D., Errecal, M., Cagnina, L., & Rosso, P. (2009) Particle swarm optimization for clustering short-text corpora. In *Proceedings of the conference on computational intelligence and bioengineering: essays in memory of Antonina Starita* (pp. 3–19)

Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern Recogn Lett, 31*(8), 651–666. https://doi.org/10.1016/j.patrec.2009.09.011

Jardine, N., & van Rijsbergen, C. J. (1971). The use of hierarchic clustering in information retrieval. *Information Storage and Retrieval, 7*(5), 217–240. https://doi.org/10.1016/0020-0271(71)90051-9

Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika, 32*(3), 241–254. https://doi.org/10.1007/bf02289588

Kulkarni, A., & Callan, J. (2010) Document allocation policies for selective searching of distributed indexes. In *Proceedings of CIKM international conference on information and knowledge management,* *association for computing machinery* (pp. 449–458) New York, NY, USA, CIKM '10, https://doi.org/10.1145/1871437.1871497

Kulkarni, A., Tigelaar, A.S., Hiemstra, D., & Callan, J. (2012) Shard ranking and cutoff estimation for topically partitioned collections. In *Proceedings of CIKM international conference on information and knowledge management, association for computing machinery* (pp. 555–564) New York, NY, USA, CIKM '12 https://doi.org/10.1145/2396761.2396833

Kummamuru, K., Dhawale, A., & Krishnapuram, R. (2003) Fuzzy co-clustering of documents and keywords. In *IEEE international conference on fuzzy systems* (Vol 2, pp. 772–777) https://doi.org/10.1109/FUZZ.2003.1206527

Larsen, B., & Aone, C. (1999) Fast and effective text mining using linear-time document clustering. In *Proceedings of ACM SIGKDD international conference on knowledge discovery and data mining, association for computing machinery* (pp. 16–22) New York, NY, USA, KDD '99, https://doi.org/10.1145/312129.312186

Le, Q., & Mikolov, T. (2014) Distributed representations of sentences and documents. In *Proceedings of international conference on machine learning* (pp. II–1188–II–1196) JMLR.org, ICML'14

Leuski, A. (2001) Evaluating document clustering for interactive information retrieval. In *Proceedings of CIKM international conference on information and knowledge management* (pp. 33–40), https://doi.org/10.1145/502585.502592

Li, C., Sun, A., & Datta, A. (2012) Twevent: Segment-based event detection from tweets. In *Proceedings of CIKM international conference on information and knowledge management* (pp. 155–164) https://doi.org/10.1145/2396761.2396785

Liu, L., Kang, J., Yu, J., & Wang, Z. (2005) A comparative study on unsupervised feature selection methods for text clustering. In *International conference on natural language processing and knowledge engineering* (pp. 597–601) IEEE https://doi.org/10.1109/NLPKE.2005.1598807

Liu, X., & Croft, W.B. (2004) Cluster-based retrieval using language models. In *Proceedings of ACM-SIGIR international conference on research and development in information retrieval, association for computing machinery* (pp. 186–193) New York, NY, USA, SIGIR '04 https://doi.org/10.1145/1008992.1009026

Liu, Y., Liu, Z., Chua, T., & Sun, M. (2015) Topical word embeddings. In *Proceedings of AAAI conference on artificial intelligence* (vol 29) https://doi.org/10.5555/2886521.2886657

Lydia, E. L., Kumar, P. K., Shankar, K., Lakshmanaprabu, S. K., Vidhyavathi, R. M., & Maseleno, A. (2018). Charismatic document clustering through novel k-means non-negative matrix factorization (KNMF) algorithm using key phrase extraction. *International Journal of Parallel Programming, 48*(3), 496–514. https://doi.org/10.1007/s10766-018-0591-9

MacQueen, J.B. (1967) Some methods for classification and analysis of multivariate observations. In: *Proceedings of the fifth berkeley symposium on mathematical statistics and probability* (Volume 1: Statistics 1:281–297)

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013) *Efficient estimation of word representations in vector space*. arXiv preprint arXiv:13013781

Modha, D. S., & Spangler, W. S. (2003). Feature weighting in k-means clustering. *Machine Learning, 52*(3), 217–237. https://doi.org/10.1023/A:1024016609528

Pal, A., & Counts, S. (2011) Identifying topical authorities in microblogs. In *Proceedings of ACM international conference on web search and data mining* (pp. 45–54) https://doi.org/10.1145/1935826.1935843

Panuccio, A., Bicego, M., & Murino, V. (2002) A hidden Markov Model-based approach to sequential data clustering. In: *Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR)* (pp. 734–743) Springer https://doi.org/10.1007/3-540-70659-3_77

Pfeifer, D., & Leidner, J.L. (2019) Topic grouper: an agglomerative clustering approach to topic modeling. In *Proceedings of ECIR european conference on IR research* (pp. 590–603) Springer https://doi.org/10.1007/978-3-030-15712-8_38

Ramage, D., DHall, Nallapati, R., & Manning, C.D. (2009) Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of conference on empirical methods in natural language processing* (pp. 248–256) https://doi.org/10.5555/1699510.1699543

Rousseeuw, P. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics, 20*, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7

Salton, G. (1971) *The SMART retrieval system—experiments in automatic document processing*. Prentice-Hall, Inc., https://doi.org/10.5555/1102022

Schütze, H. (1992) Dimensions of meaning. In *Proceedings of the 1992 ACM/IEEE conference on supercomputing* (pp. 787–796) IEEE Computer Society Press, Washington, DC, USA, Supercomputing '92

Shafiei, M., Wang, S., Zhang, R., Milios, E., Tang, B., Tougas, J., & Spiteri, R. (2007) Document representation and dimension reduction for text clustering. In *Proceedings of IEEE international conference on data engineering* (pp. 770–779) IEEE https://doi.org/10.1109/ICDEW.2007.4401066

Song, W., & Park, S.C. (2006). Genetic algorithm-based text clustering technique: Automatic evolution of clusters with high efficiency. In *Proceedings of WAIMW seventh international conference on web-age information management workshops*

Spirin, N., & Han, J. (2012). Survey on web spam detection: principles and algorithms. *SIGKDD Explorations, 13*(2), 50–64. https://doi.org/10.1145/2207243.2207252

Thorndike, R. L. (1953). Who belongs in the family? *Psychometrika, 18*(4), 267–276. https://doi.org/10.1007/BF02289263

Tomasini, C., Emmendorfer, L., Borges, E.N., & Machado, K. (2016) A methodology for selecting the most suitable cluster validation internal indices. In *Proceedings of ACM symposium on applied computing* (pp. 901–903) https://doi.org/10.1145/2851613.2851885

Tomašev, N., & Radovanović, M. (2016) *Clustering evaluation in high-dimensional data* (pp. 71–107) Springer International Publishing https://doi.org/10.1007/978-3-319-24211-8_4

Tombros, A., Villa, R., & van Rijsbergen, C. J. (2002). The effectiveness of query-specific hierarchic clustering in information retrieval. *Information Processing&amp; Management, 38*(4), 559–582. https://doi.org/10.1016/S0306-4573(01)00048-6

van Rijsbergen, C. J. (1979). *Information Retrieval*. Butterworths.

Voorhees, E. M. (1986). Implementing agglomerative hierarchic clustering algorithms for use in document retrieval. *Information Processing and Management, 22*(6), 465–476. https://doi.org/10.1016/0306-4573(86)90097-X.

Voorhees, E. M., & Harman, D. K. (Eds.). (2005). *TREC experiment and evaluation in information retrieval*. The MIT Press.

Weber, R., Schek, H., & Blott, S. (1998) A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In *Proceedings of VLDB international conferemce on very large databases* (vol 98, pp. 194–205)

Wei, X., & Croft, W.B. (2006) LDA-based document models for ad-hoc retrieval. In *Proceedings of ACM-SIGIR international conference on research and development in information retrieval* (pp. 178–185) https://doi.org/10.1145/1148170.1148204

Willett, P. (1988). Recent trends in hierarchic document clustering: a critical review. *Information Processing&amp; Management, 24*(5), 577–597. https://doi.org/10.1016/0306-45738890027-1

Xu, J., & Croft, W.B. (1999) Cluster-based language models for distributed retrieval. In *Proceedings of ACM-SIGIR international conference on research and development in information retrieval, association for computing machinery* (pp. 254–261) New York, NY, USA, SIGIR '99 https://doi.org/10.1145/312624.312687

Xu, W., Liu, X., & Gong, Y. (2003) Document clustering based on non-negative matrix factorization. In *Proceedings of ACM-SIGIR international conference on research and development in information retrieval* (pp. 267–273) https://doi.org/10.1145/860435.860485

Yang, K., & Miao, R. (2018) Research on improvement of text processing and clustering algorithms in public opinion early warning system. In *International conference on systems and informatics*https://doi.org/10.1109/ICSAI.2018.8599424

Zhang, W., Tang, X., & Yoshida, T. (2015). TESC: an approach to TExt classification using semi-supervised clustering. *Knowledge-Based Systems, 75,* 152–160.

Zobel, J. (1998) How reliable are the results of large-scale information retrieval experiments? In *Proceedings of ACM-SIGIR international conference on research and development in information retrieval* (pp. 307–314) https://doi.org/10.1145/290941.291014

Zobel, J., & Moffat, A. (2006) Inverted files for text search engines. *ACM Computing Surveys 38*(2), 6–es, https://doi.org/10.1145/1132956.1132959

Zobel, J., Moffat, A., & Park, L. (2009). Against recall: is it persistence, cardinality, density, coverage, or totality? *Proceedings of ACM-SIGIR International Conference on Research and Development in Information Retrieval, 43*(1), 3–8. https://doi.org/10.1145/1670598.1670600