Evaluation of text clustering methods and their dataspace embeddings: an exploration

Alain Lelu, Martine Cadot

EVALUATION OF TEXT CLUSTERING METHODS AND THEIR DATASPACE EMBEDDINGS: AN EXPLORATION.

Alain Lelu, rtd, Université de Bourgogne-Franche-Comté[1]

Martine Cadot, LORIA[2]

Abstract

Fair evaluation of text clustering methods needs to clarify the relations between 1)pre-processing, resulting in raw term occurrence vectors, 2)data transformation, and 3)method in the strict sense. We have tried to empirically compare a dozen well-known methods and variants in a protocol crossing three contrasted open-access corpora in a few tens transformed dataspaces. We compared the resulting clusterings to their supposed "ground-truth" classes by means of four usual indices. The results show both a confirmation of well-established implicit combinations, and good performances of unexpected ones, mostly in spectral or kernel dataspaces. The rich material resulting from these some 600 runs includes a wealth of intriguing facts, which needs further research on the specificities of text corpora in relation to methods and dataspaces.

1 - Introduction : motivations and goals.

Evaluation of text clustering methods is one of the key issues in the problem of bibliometric delineation of scientific fields. As co-authors of [Zitt et al. 2019] we have tried to test seventeen clustering methods with a publicly available real-life test set, the Reuters' test bench [Lewis et al. 2004] which adds up several difficulties of text clustering, i.e. mainly strongly unbalanced man-made classes (the targeted “ground-truth”), and texts of unbalanced sizes. Our report is accessible online [Cadot, Lelu 2018] as a supplementary material to the above-mentioned book chapter. An unexpected result was that antique agglomerative methods, especially Ward hierarchical clustering, performed better than many more recent ones. Was it the case for all types of corpora? Above all we realized that for the sake of fair comparisons, as well as conceptual clarity, we should clearly separate the transformations of the raw word-count data (for example into Salton tf-idf vector representation, or Laplacian spectral space, etc.) from the algorithms in the strict sense, instead of using long-time accepted implicit combinations. For example, no rational reasons forbid using e.g. Non-negative Matrix Factorization in a spectral space. This consideration is in line with the conceptual clarifications operated in [Van Mechelen et al. 2018]; last, but not least, unexpected recommendations may proceed from non-classic combinations. This clarification is one of our guiding threads in the present research.

Though restricting our scope to text clustering, it is clear that many types of texts need now to be processed: abstracts or plain texts of scientific papers, which are our primary scientific interest, or journal, litterary or legal texts, or texts originating in the social nature of Internet communications, such as contributions to forum discussions, or social networks. We decided to base our present survey on three typical and contrasted test sets: a full-text scientific database, a wire of press agency, and an Internet discussion forum. It is clear that the complete text preprocessing chain is out of the goal of our research, so we have to rest on one same linguistic – or weakly linguistic – term, lemma

[1] Alelu@orange.fr

[2] Martine.Cadot@loria.fr - corresponding author

or stem extraction scheme, and same elimination of infrequent or too frequent words. This point must not keep us from exploring the influence of truncating the remaining vocabulary in chosen distribution quantiles. Usual benchmark studies mention an absolute occurrence threshold, period... All these specifications led us to the choices we expose in the methodology section.

Of course the options on methods and types of dataspaces to be considered are inevitably somehow arbitrary: we tried to take account of the most usual algorithms, or method families, such as K-means, hierarchical agglomerative clustering, spectral clustering, graph clustering, kernel clustering, and we added two more specific methods, i.e. Non-negative Matrix Factorization (NMF) and Latent Dirichlet Allocation (LDA), which amounts to a dozen methods and variants.

Concerning dataspaces, we chose to add to the plain term occurrence vector space the transformed spaces by Salton's and Okapi's tf-idf weighting schemes, by chi-square metrics, by Laplacian spectral decomposition, by Correspondence Analysis, and last by order-2 polynomial kernel expansion. These transformations are referenced and formally presented in the next section.

Given the combinatorics of the three main elements – text types, dataspaces and algorithms – our research study could be nothing but an exploration, strongly constrained by available resources. However, some interesting conclusions will be drawn out this exploration. In the final conclusion, we will deal with what may be continued and deepened in our perspective, given the results.

Let us close this introduction saying that we are indebted to the remarkable initiative of the Brazilian LABIC team [Rossi et al. 2013] who homogeneously pre-processed [LABIC stemmer] some forty text collections and made the document by term matrices available online on their site [LABIC data].

2 - Methodology

Drawn from our experience of clustering methods, our first claim is that using the supervised learning methodology for comparing non-supervised methods is in any case better than relying on unfounded claims or comparisons to be suspected of author's bias. In relation to the issue of author's bias, let us state that, though being authors of a few clustering algorithms (Axial K-means, Local Component Analysis, Germen), we have excluded these algorithms of our survey. This supervised methodology may at least result, in default of a universal ranking of methods, in fruitful reflections on the typology of texts, or the nature of the human categorization and abstraction process and its similarities and differences with methods mostly optimizing an intrinsic objective function.

Another core imperative we have set is transparency and reproducibility: in addition to the direct link to the document-by-term matrices we have provided above, a complementary material HAL site [HAL CNRS] will gather the data and code we used. Though most algorithms are theoretically insensitive to the ordering of input vectors, in practice we experienced that tied effects, among others, could affect the results. This is why we have randomly scrambled the data vectors, and will drop our scrambled data and label files on the above-cited site.

1.1 - Choice of test corpora

The three "prototype" test corpora mentioned in the introduction are, first, Reuters' "ModApté Split" [Apté et al. 1994] limited to the eight most important classes ("Re8" in the present study, 7674 documents,8901 terms), second, the ACM collection made of the proceedings of forty conferences in different computer science areas (3493 papers, 60 768 terms), third, the "20 Newsgroups" collection ("Ng20") composed of 18 808 messages posted in twenty Usenet groups (45 434 terms). The size of the man-made reference classes is strongly unbalanced in the case of Re8 (two of them constitute 81% of the documents), roughly equal in the case of ACM and Ng20.

It is to be noted that the sole Reuters' class labels are issued from a direct manual indexing. The two others origin in the concatenation of sub-corpora of comparable size. They could therefore be considered as "semi-real-world data", not really representative of real-life non-annotated corpora. Note also that the mean "Silhouette" coefficient [Kogan, 2007], which measures the inter-class contrasts, is higher (.89) in the case of Re8 than in Ng20 (.80) and ACM (.75) ones.

1.2 - Truncating the vocabularies

As the size of the vocabularies are unbalanced (Re8: about 8900 terms, ACM: 60 800 terms, Ng20: 45 000 terms) but extensive (the hapaxes, i.e. terms of total occurrence one, are included in this count), we decided a common scheme for a vocabulary-independent truncation by thresholds: in addition to the basic option of retaining the whole vocabulary, we built two sub-corpora per test corpus retaining the third quartile of the term distribution (25% of the total occurrences), and the seventh "octile" (12.5%).

1.3 - Choice of clustering methods

We have affected a lower priority to algorithms with two parameters (DbScan, Affinity Propagation, Smart Local Moving Algorithm) or one parameter with deceptive results on Reuters' corpus (Density Peaks, Independent Component Analysis, Fuzzy c-means, K-Means++). We selected:

- First, plain *K-means clustering* ("KM"), see [Mac Queen 1967] algorithm (in the adaptive case) and [Forgy 1965] (in the usual iterative case), initialized by randomly drawing data vectors - we had an unconvincing experience of K-means++ initialization in the context of text clustering. We implemented 20 elementary runs, or "passes", per run, selecting the "best" one in terms of the local optimum of the K-means objective function (sum of squared intra-cluster Euclidean distances).

- *Hierarchical agglomerative clustering* with two linkage variants: average link ("HCa"), and Ward ("HCw") [Ward 1963]. Originally in O(#documents^3) time complexity, the more recent contributions [Murtagh 1984] and [Müllner 2013] have lowered this constraint to O(#documents^2).

- *Spectral clustering* [Meila, Shi 2000]: we used the "standard" combination K-means/Laplacian spectral dataspace, but also explored (with success, see below) many other combinations.

- *Graph clustering methods*: we chose the two most broadly recognized ones, i.e. *Louvain* [Blondel et al. 2008] and *InfoMap* [Rosvall, Bergstrom 2007]. Note that these methods, in contrast to all the other tested ones, do not need fixing a desired number of clusters, hence a major operational advantage when no idea of the "true number of clusters" is known beforehand - hierarchical clustering being in an intermediate position, as in one run it leaves the choice of the cluster number to the user.

- *Non-negative Matrix Factorization* ("NMF") [Lee, Seung 1999]: this decomposition is akin to be used as a clustering method, when the label of a document is attributed depending on the axis of its maximum projection. As this method converges to local optima of its objective function, we implemented the same "20 runs" strategy as for K-means. Note that the resulting gradual representation of documents in the clusters is also interesting in that outliers, or even additional cluster seeds, may be identified - one kind of "possibilistic" clustering.

- *Latent Dirichlet Allocation* ("LDA") [Blei et al. 2003] is well-known and much respected as deeply founded in theoretic grounds.

- *Kernel clustering* [Girolami 2002]: thanks to the "kernel trick", a document by document similarity matrix ("Gram matrix") is built without explicit expansion of the raw dataspace by a kernel function. Here we used an order-2

polynomial kernel, which amounts to take into account the wholeness of the 2-term itemsets in each document when comparing one to another (in addition to standard "1-term" itemsets). In this case the raw dataspace is not made of numeric occurrence vectors, but of binary existence ones.

1.4 - Choice of dataspaces:

In addition to the plain term-occurrence vector space, we have considered and built:

- *Salton's vector space*, weighted by the classic tf-idf scheme:

x_{ij} = tf idf

where tf = n_{ij}

idf = log n - log $e_{.j}$

> **Notations: N** is the document by-term-matrix, n_{ij} is the occurrence number of term j in document i, $n_{i.}$ is the total occurrences of terms in document i, $n_{.j}$ is the total occurrences of term j, n.. is the grand total of occurrences in the corpus, n is the number of documents.
>
> **E** is the binary document by term matrix, e_{ij} is 1 if term j exists in document i, 0 else, $e_{i.}$ is the total number of terms which appear in document i, $e_{.j}$ is the number of documents in which term j appears, e.. is the grand total of term appearances in the corpus.
>
> k* is the number of manually attributed classes in each corpus.

- *Okapi* (also coined BM25) *vector space* [Robertson et al. 1994], with a more cryptic, but statistically grounded, weighting scheme [Robertson et al. 1994]:

x_{ij} = tf idf

tf = n_{ij} (a+1) / (n_{ij} + a(1-b+b($n_{i.}$n/n..)) ; recommended values: a=1.2 b=.75

idf = log(n - $e_{.j}$ + .5) - log($n_{i.}$ + .5)

For enabling idf to be always positive or zero, we have ceiled the value ($e_{.j}$+$n_{i.}$) to n.

- *Chi-square metrics*, which amounts to a Euclidean vector space with vectors transformed as suggested in [Legendre, Gallagher 2001]:

$x_{ij} = n..^{1/2}\ n_{ij} / (n_{i.}\ n_{.j}^{1/2})$

- *Laplacian spectral space* [Von Luxburg 2007]:

Given **Cos**, the (truncated or not) cosine matrix between the documents, which defines a non-oriented graph, and **s** the sum vector of its rows or columns, the transformed matrix

$\mathbf{Q_{lapl}}$ = diag($\mathbf{s}^{-1/2}$) **Cos** diag($\mathbf{s}^{-1/2}$) gives rise to a Singular Value Decomposition (SVD):

$\mathbf{Q_{lapl}} = \mathbf{U_{lapl}}\ \mathbf{D}\ \mathbf{V_{lapl}}$

The first rank factors concentrate the relevant information. When aiming at k* classes, we chose to take into account three configurations of $\mathbf{U_{lapl}}$, with respectively integer(k*/2), k* and 2k* top-ranking factors.

- *Correspondence Analysis spectral space* [Benzecri 1973] [Greenacre 1984] [Lebart et al. 1998]:

The transformed matrix $\mathbf{Q_{ca}}$ = diag{$n_{i.}^{-1/2}$} **N** diag{$n_{.j}^{-1/2}$} gives rise to a SVD decomposition

$\mathbf{Q_{ca}} = \mathbf{U_{ca}}\ \mathbf{L}\ \mathbf{V_{ca}}$

resulting in the CA factors:

$\mathbf{F} = n..^{1/2}\ \mathrm{diag}\{n_{i.}^{-1/2}\}\ \mathbf{U_{ca}}\ \mathbf{L}$

As the first factor is trivial, we chose to take into account three configurations, with respectively 1+integer(k*/2), 1+k* and 1+2k* top-ranking factors.

Note that Euclidean distances in the complete factor space equal chi-square distances [Benzécri 1973]. Therefore, truncating this space in this manner amounts to consider "partial chi-square" distances*, a priori* more relevant than chi-square distances. We will check this point below.

These six transformations of a document-by-term matrix are convenient for the KM, NMF, LDA and Spectral Clustering methods. Other methods, such as Hierarchical Clustering, Graph methods and Kernel methods, need a document-by-document similarity (or dissimilarity) matrix. We have derived the distance or cosine matrices for all the above-mentioned dataspaces. Depending on each dataspace-method combination, we have used Euclidean distance or "cosine" distance (i.e. 1-cosine, which weights half of the squared chord distance [Legendre, Gallagher 2001]).

- *Kernel space* [Girolami 2002]:

In matrix notations and in the case of order-2 polynomial kernel, the Gram similarity matrix writes:

$\mathbf{X_{k2}} = (\mathbf{1} + \mathbf{E}\ \mathbf{E}')^{.\wedge 2}$ where $\mathbf{X}^{.\wedge 2}$ means squaring each value of matrix $\mathbf{X}$

Given the much contrasted values in this matrix, the cosine distance is well-fit to this dataspace.

1.5 - Choice of evaluation measures

We chose the four most usual indices encountered in the evaluation literature, i.e. first, Normalized Mutual Information (NMI) [Cover, Thomas 1991] and Adjusted Rand Index (ARI) [Rand 1971], which compute independently from the number and labels of clusters; and second, mean local class-vs.-cluster F-scores (F) [Van Rijsbergen 1979] and global Purity score (i.e. 1-global error rate) which need the same number of clusters and classes, and same labels. Even when this was not the case (graph methods), we have aligned the k classes and the k most "analogue" clusters, in the sense of local F-scores, by means of the ranking issued from the first non-trivial factor in the Correspondence Analysis of the classes by clusters F-score matrix.

1.6 - Code implementation and computer efficiency

As computer efficiency is out of our goals, we implemented the data transformations, method code, and post-processing code in an Octave environment, on an Intel 6-core I7, 3.33GHz, 48Go RAM computer. Method codes were derived from existing Matlab® codes (links to the original pieces of code are available in the supplementary material). Their degree of computing time optimization varies considerably: e.g. in the case of the 19 000 documents Ng20 test set, from 2 minutes for twenty elementary runs of the standard "litekmeans.m" code, to 6 hours for one run of Louvain method.

2 - Evaluation results for each method

We have tried as much as possible to cross-combine corpora × data transformations × methods. This was not always possible, due to constraints such as computing time or resources devoted to systematically poor results.

2.1 - K-means

Altogether with hierarchical agglomerative clustering, this method is an inescapable classic of exploratory data analysis. Our results confirm why it is so widely used till now: good performances (nmi>.60 on ACM and Ng20 corpora), low computing complexity O(#documents × #terms), fast runs as noted above.

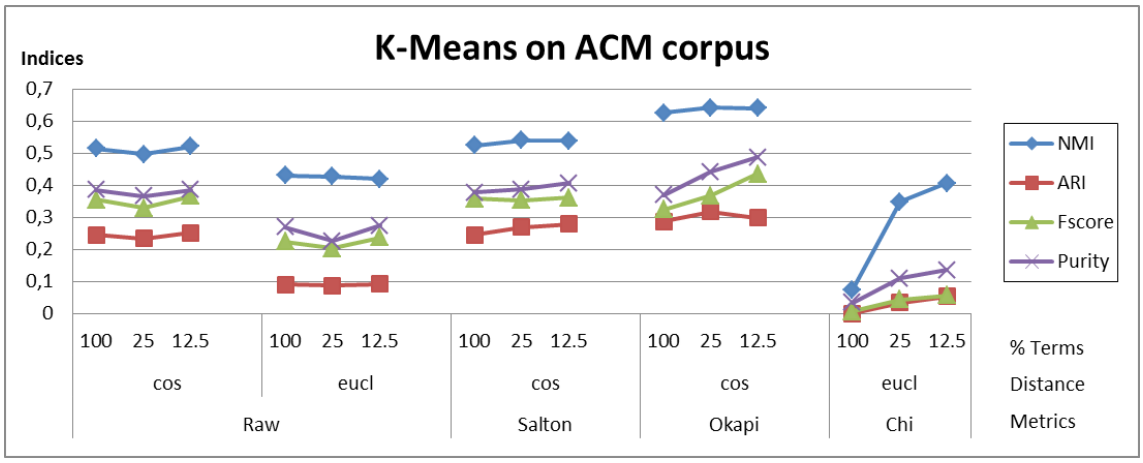

Figure 1: K-means on ACM corpus

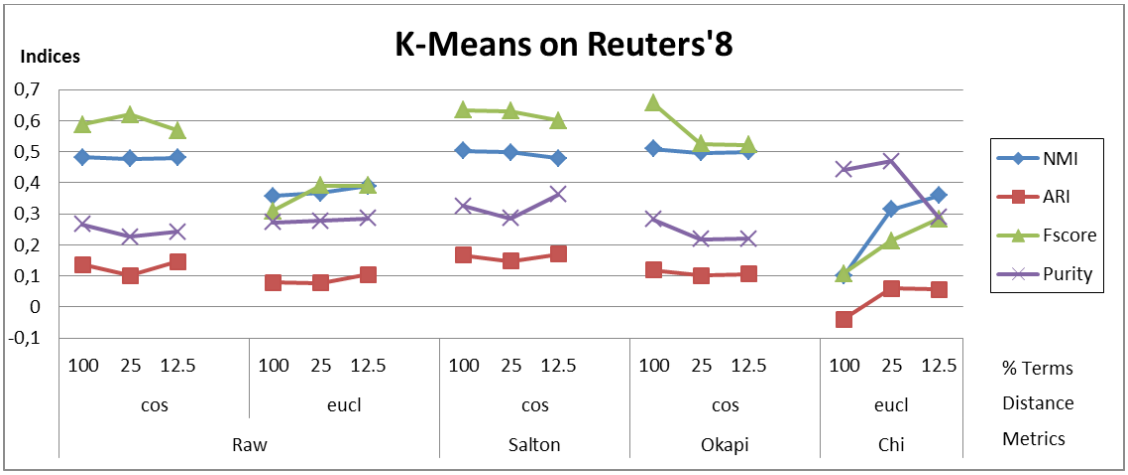

Figure 2: K-means on Re8 corpus

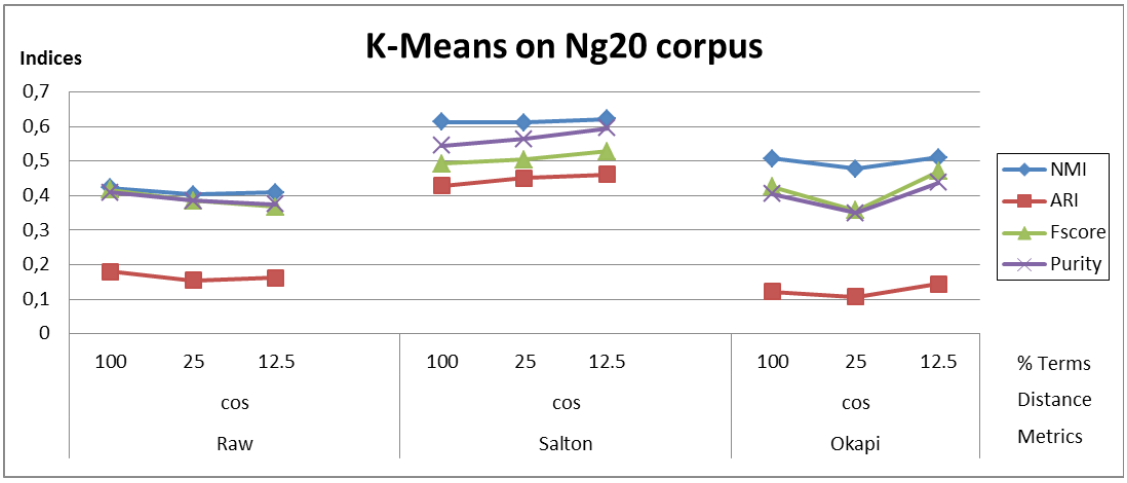

Figure 3: K-means on Ng20 corpus

A remark about evaluation indices: in the case of the two balance-sized corpora, they both globally go much in the same line - a bit more uneven in the case of Re8.

The vocabulary truncation does not seem to be quite influential, except in the case of the Chi-square space. So one recommendation may be drawn: a 12,5% vocabulary threshold seems good enough for acceptable K-means results, resulting in a lighter processing task.

Examining performances, the results are surprising: Okapi dataspace performs best for ACM, Salton's space for Ng20, and both perform equally for Re8, as well as raw occurrence dataspace. Conclusions would need a specific study of the peculiarities of text corpora with regard to these disorientating kinds of observations.

2.2 - NMF

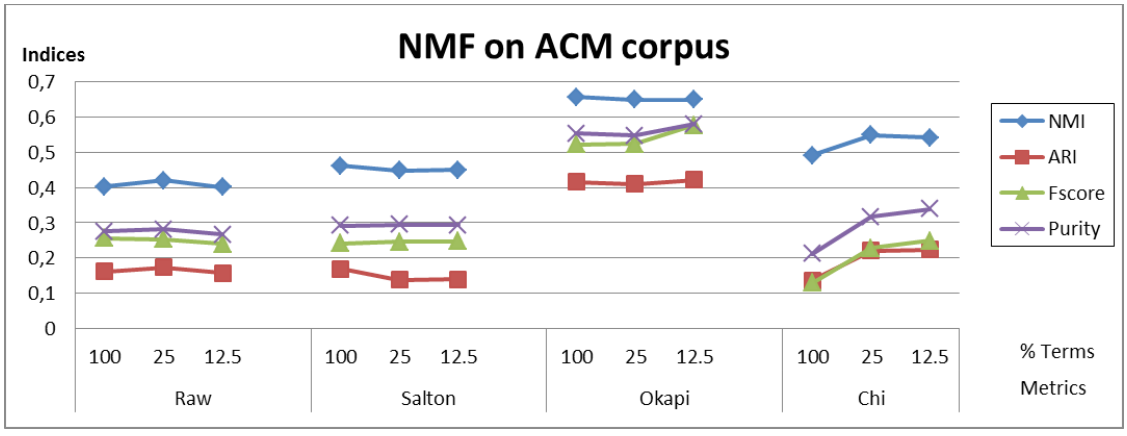

Figure 4: NMF on ACM corpus

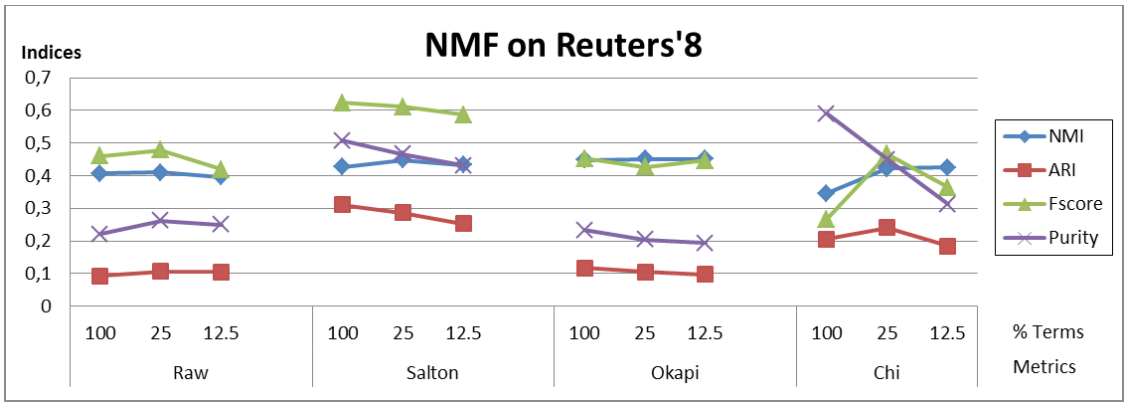

Figure 5: NMF on Re8 corpus

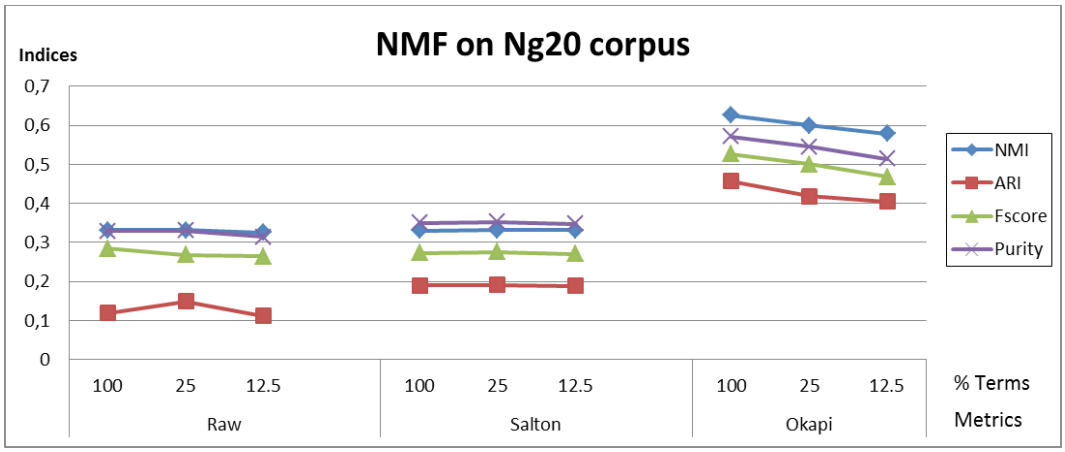

Figure 6: NMF on Ng20 corpus

This more recent method confirms the surprising observation above: ACM and Ng20 yield good performances in Okapi space, whereas Re8 does in Salton's space.

2.3 - LDA

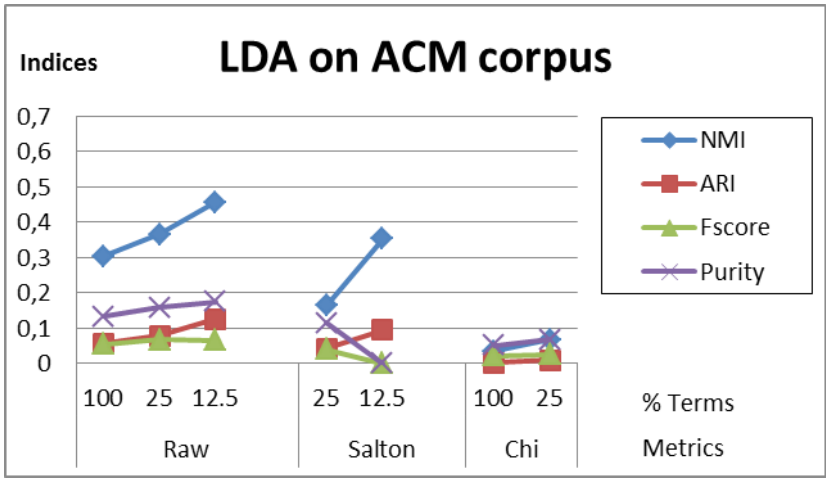

Figure 7: LDA on ACM corpus

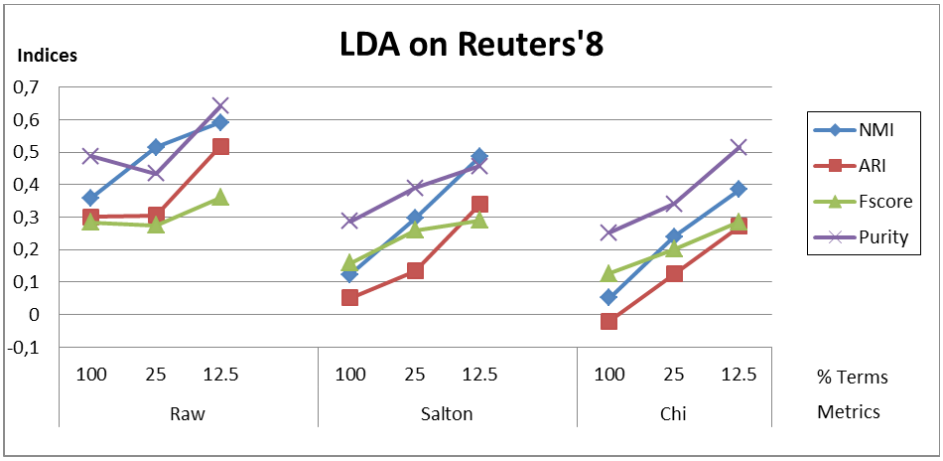

Figure 8: LDA on Re8 corpus

The somehow chaotic performances and the unpredictable computing time (10 to 40 minutes) led us to not process the 3-times bigger Ng20 corpus.

However, it seems that LDA is more fit to the raw Euclidean dataspace, perhaps more in line with its underlying statistical hypotheses. Another observation: the more vocabulary is truncated, the better.

2.4 - Linkage methods

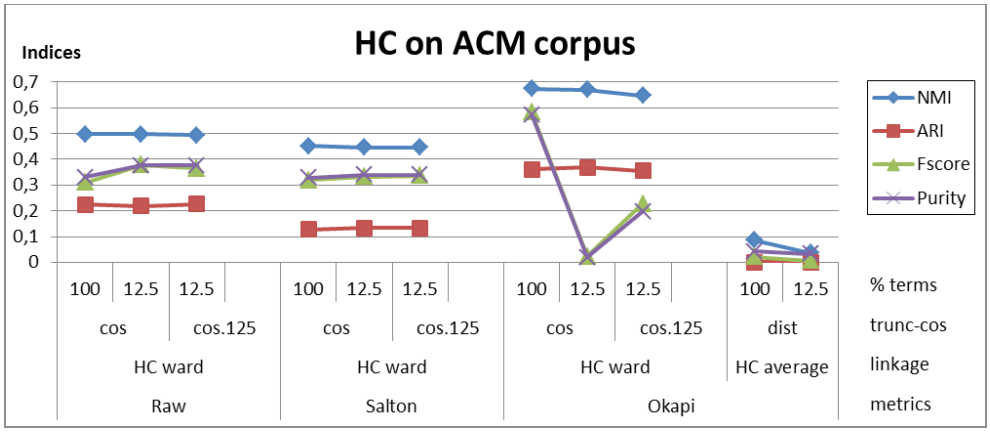

Figure 9: Hierarchical Clustering on ACM corpus

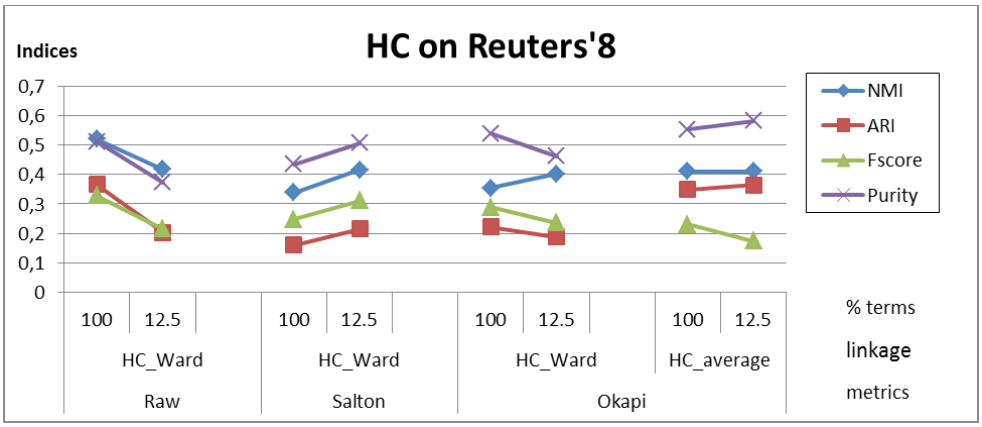

Figure 10: Hierarchical Clustering on Re8 corpus

At first glance, it seems that the average link variant performs correctly with Reuters'8, but not with ACM - another intriguing fact. Ward linkage variant seems also a clear winner.

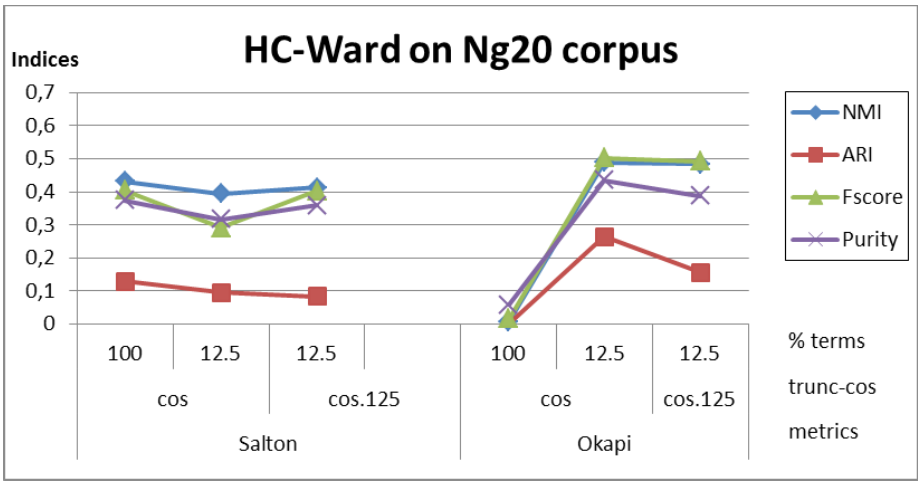

Figure11 - Hierarchical Clustering on Ng20 corpus

Using Okapi dataspace outperforms Salton's and Raw ones for ACM and Ng20 corpora, in line with the observations above.

2.5 - Spectral clustering

ACM corpus

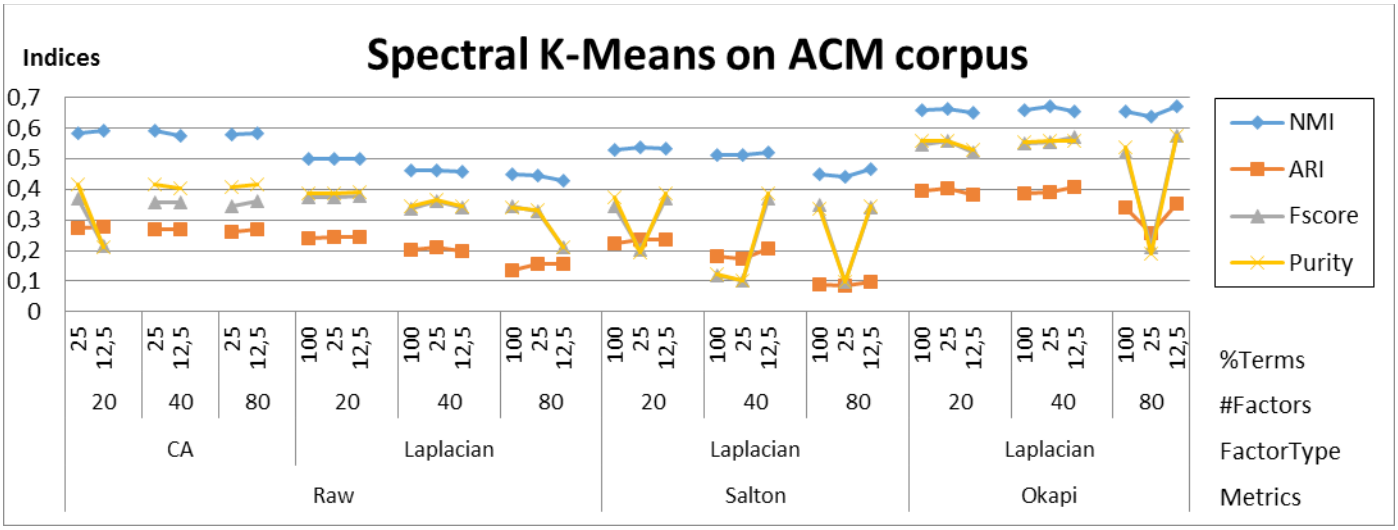

Figure 12: Spectral K-means Clustering on ACM corpus

Note the excellent performance of this method in Okapi-transformed dataset with Laplacian spectral extraction, whatever the number of factors and term percentage, and the good performance in Correspondence Analysis factor space.

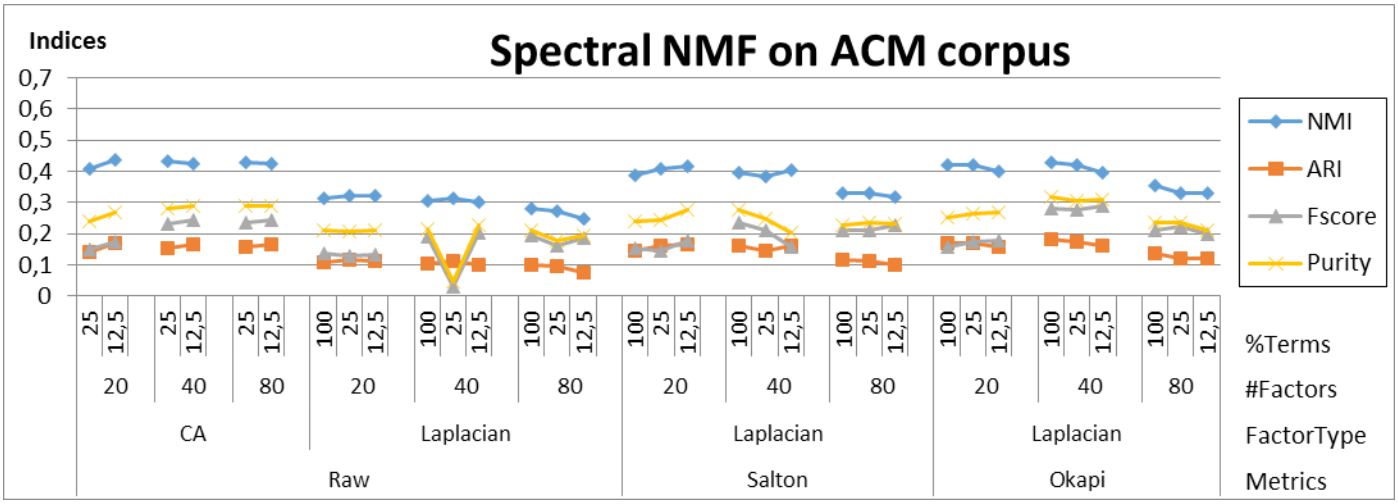

Figure 13: Spectral Non-negative Matrix Factorization clustering on (un-)weighted ACM corpus

NMF is good, but not as competitive as K-Means in the same spectral context.

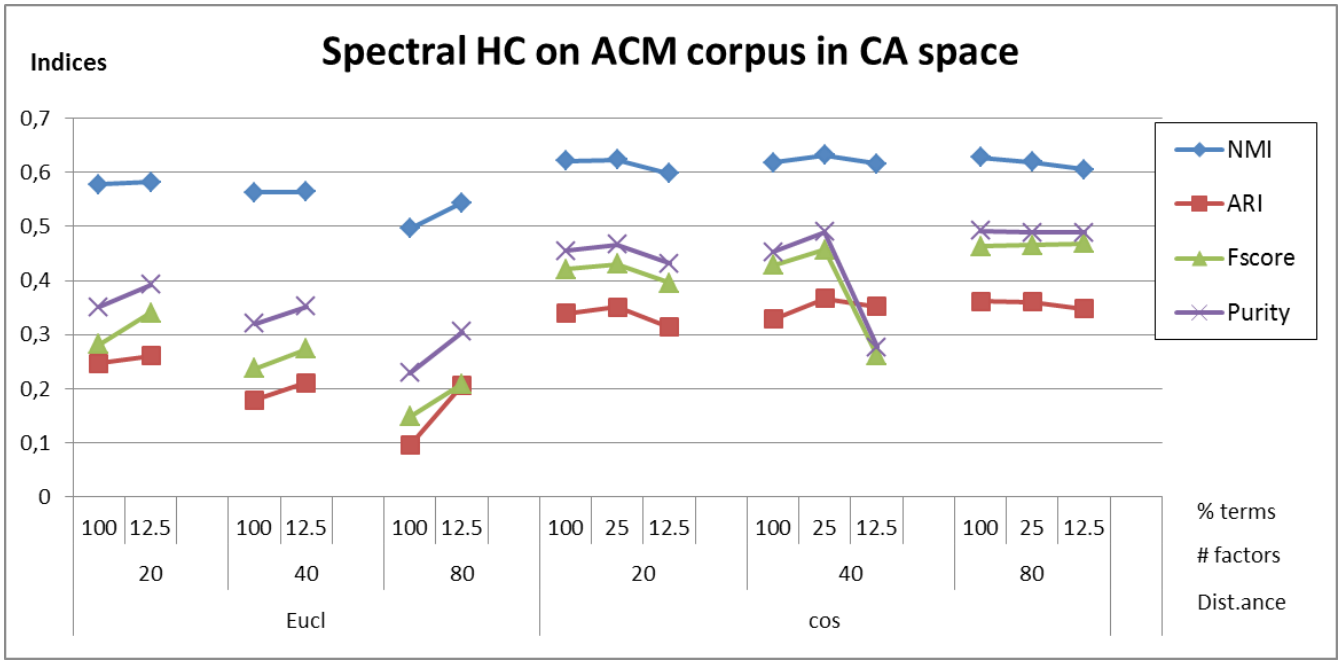

Figure 14: Spectral Hierarchical-Ward Clustering on ACM corpus in Correspondence Analysis factor space (cosine distance)

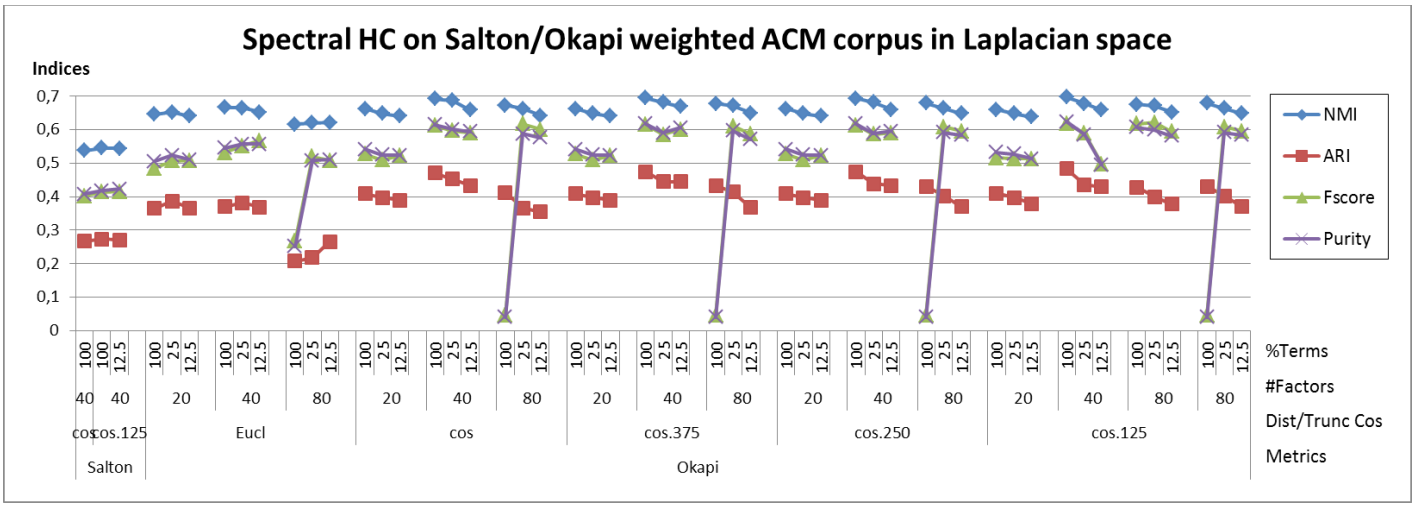

Figure 15: Spectral Hierarchical-Ward Clustering on ACM corpus in Okapi/Salton weighted Laplacian factor space (quantile-truncated –or not– cosine distances)

An unexpected result is that spectral HC family seems to outperform all other methods tested yet on the ACM corpus.

Re8 corpus

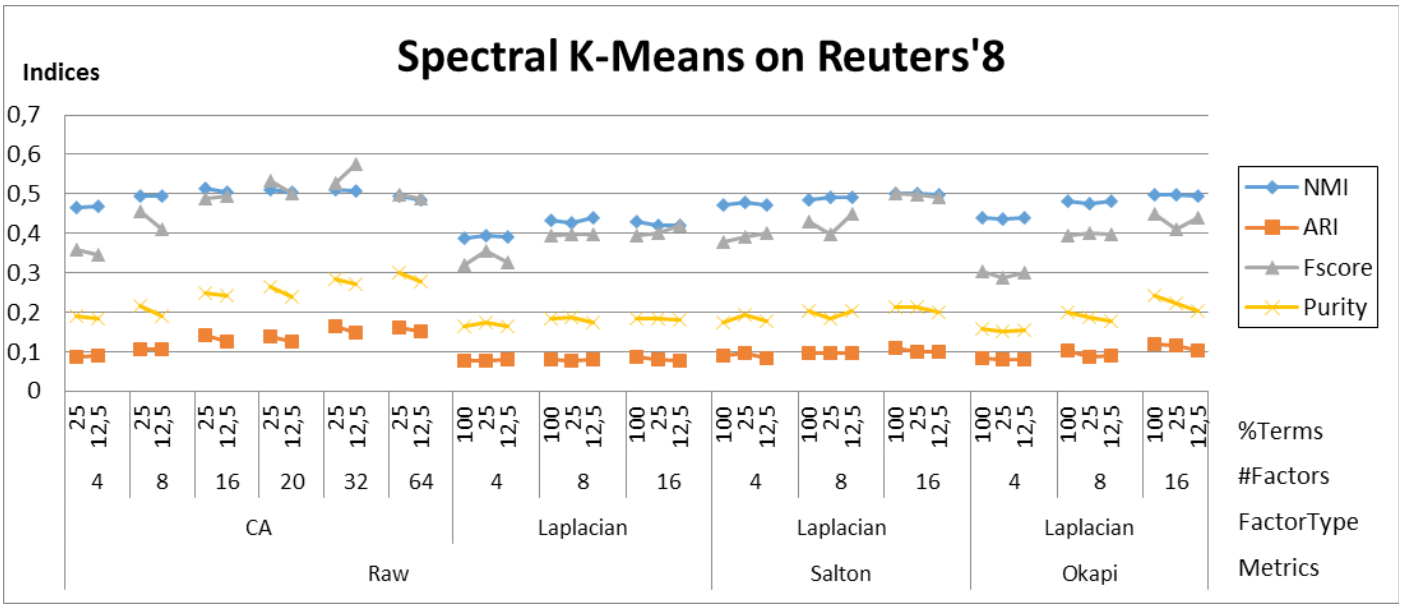

Figure 16: Spectral K-means Clustering on Reuters'8 corpus

"Attraction" of Reuters'8 corpus towards Raw and Salton's dataspaces in confirmed, as well as a relative insensitivity to term truncation and dimension of the factor space.

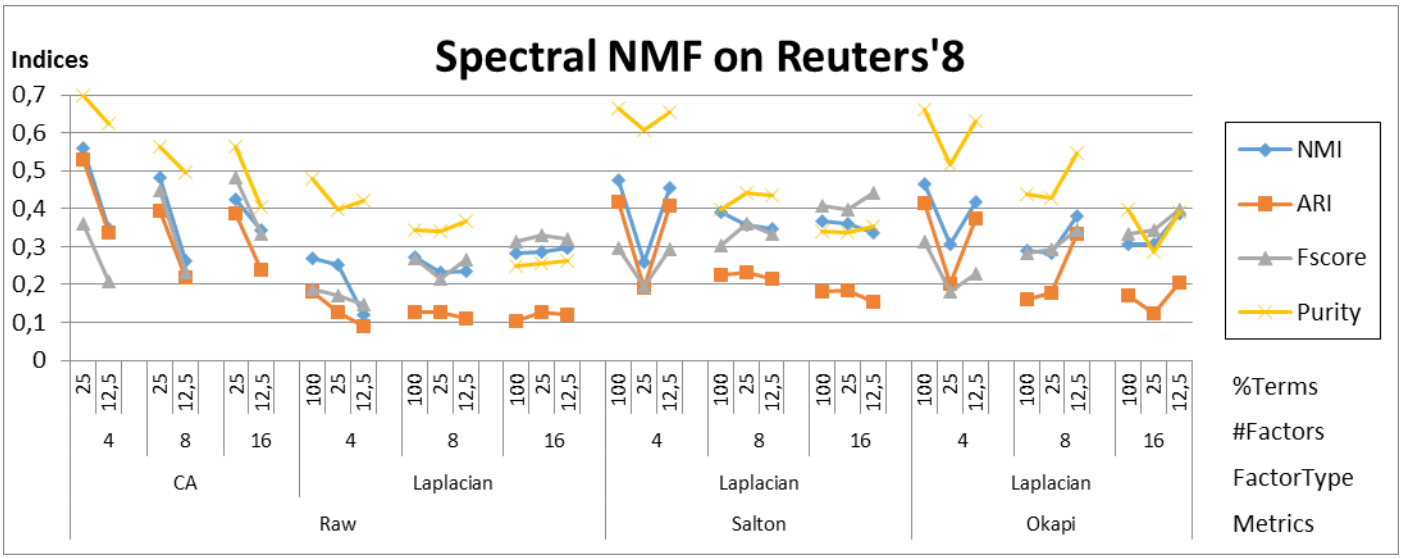

Figure 17: Spectral NMF Clustering on Reuters'8 corpus

Not so good, purity is privileged, seemingly due to its good performance in the two major clusters.

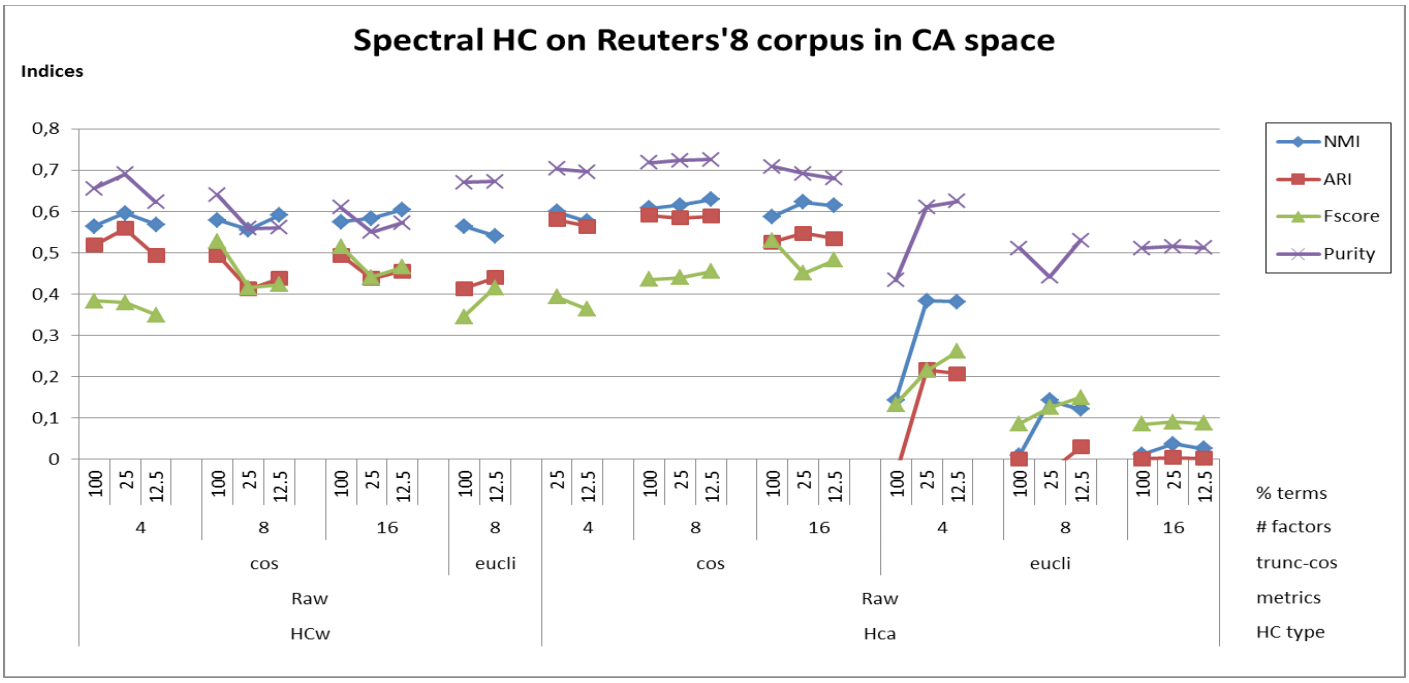

Figure 18: Spectral Hierarchical Clustering on Re8 corpus in Correspondence Analysis factor space (cosine distance)

HC-average is clearly a bad option in raw CA space, but a very good one using cosine distance in this dataspace.

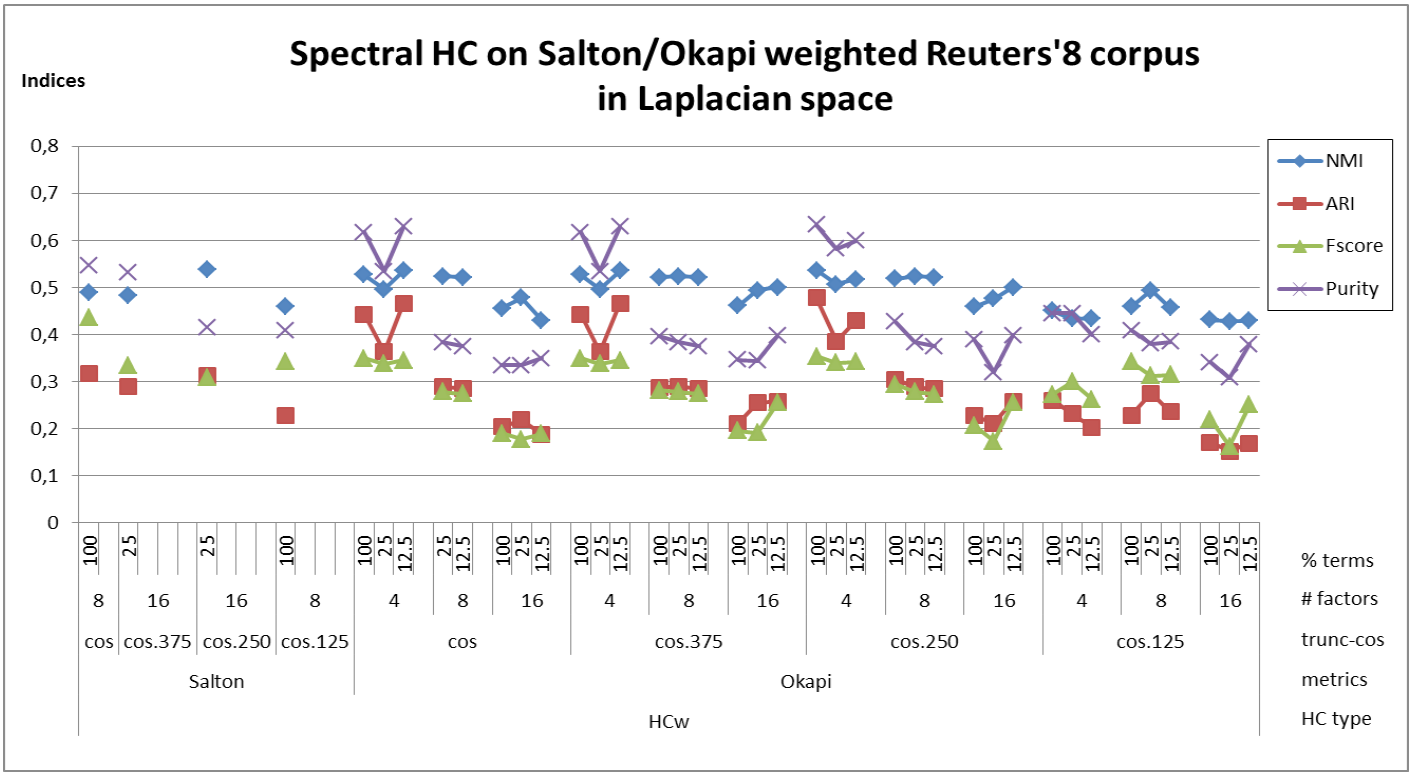

Figure 19: Spectral Hierarchical-Ward Clustering on Re8 corpus in Okapi/Salton-weighted Laplacian factor space (quantile-truncated –or not– cosine distances)

Ng20 corpus

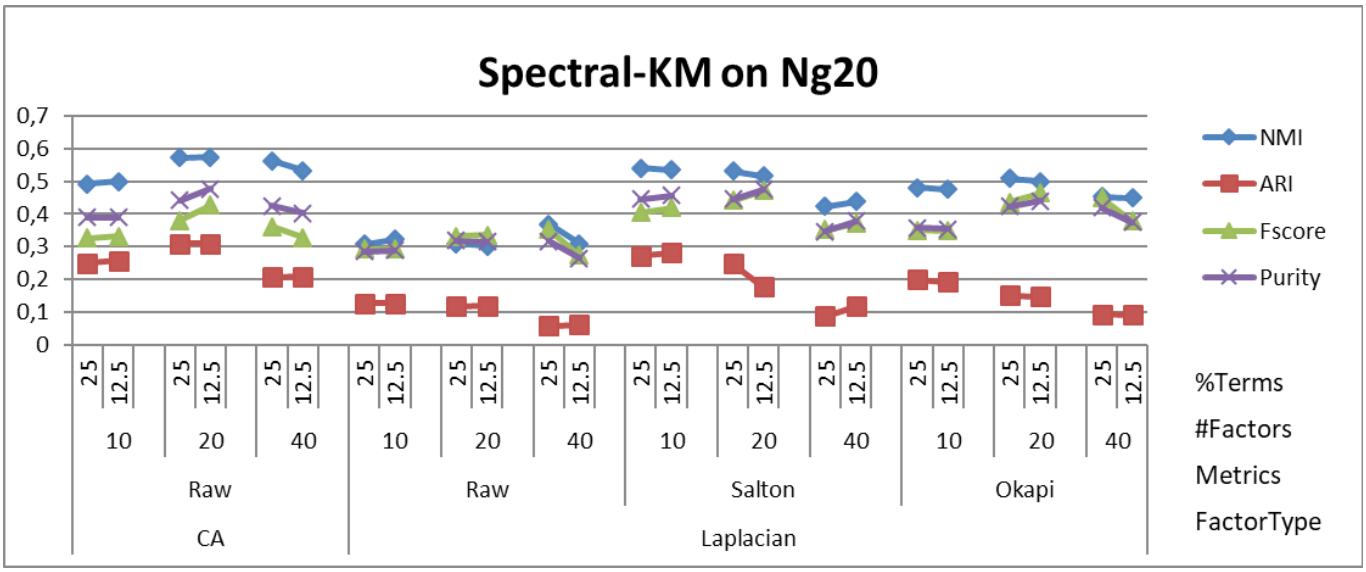

Figure 20: Spectral K-means Clustering on Ng20 corpus

Excellent performance in k*-factors Correspondence Analysis dataspace, slightly better than in Laplacian ones.

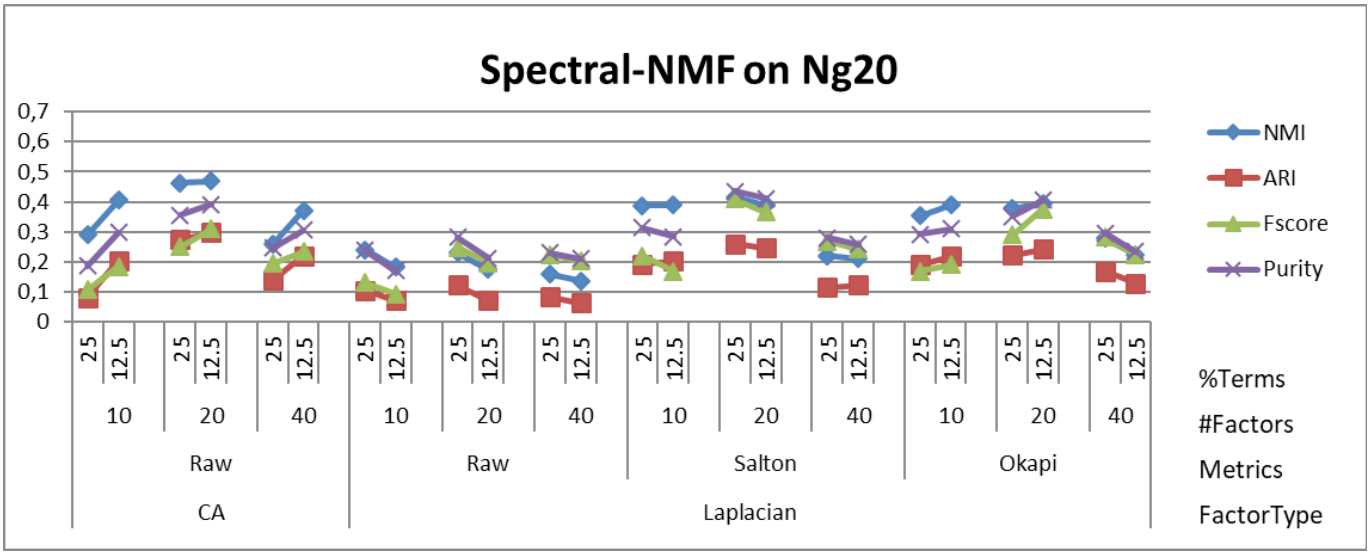

Figure 21: Spectral NMF Clustering on Ng20 corpus

Does not add benefits to other combinations. NMF seems more fit to operate in non-spectral dataspaces (Okapi, Salton, Kernel - see above) than in spectral ones: a major difference with K-Means, "comfortable" in all contexts.

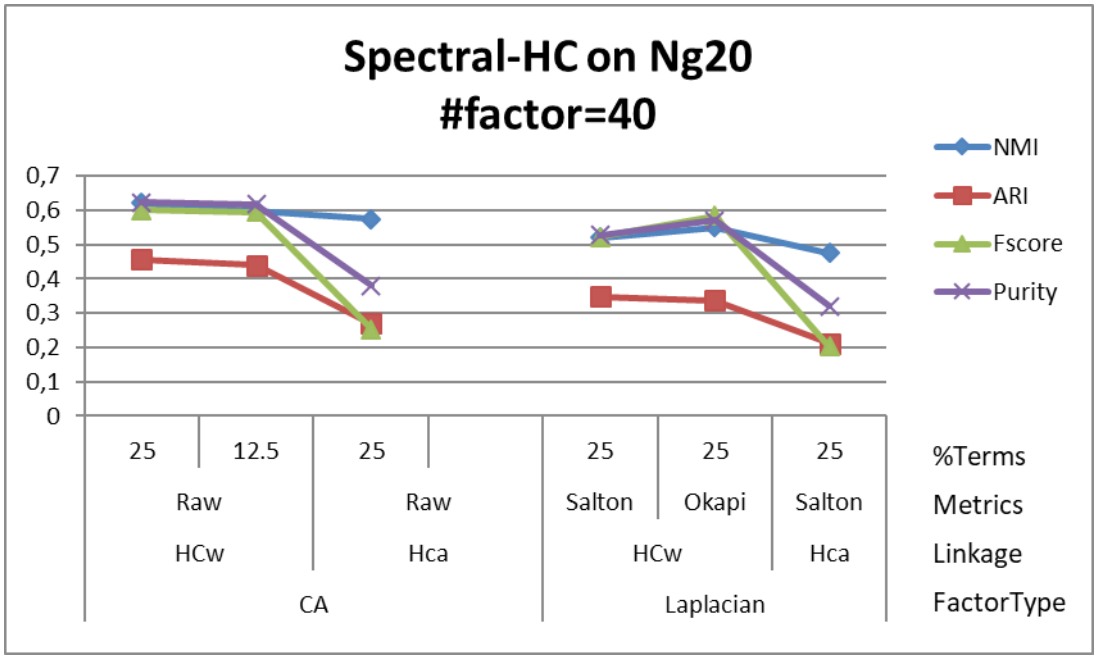

Figure 22: Spectral Hierarchical Clustering on Ng20 corpus in various factor spaces (cosine distances)

Spectral Hierarchical Clustering with Ward linkage option clearly outperforms average linkage one, all the more so as average linkage option is far more time-consuming.

2.6 - Kernel clustering

When using a second-order polynomial kernel, this more recent method is not based on term counts, unlike all other methods, but on binary presence/absence vectors, which makes it possible to take into account term 2-itemsets, i.e. supersets of term bigrams. As semantic units seem to be embedded in word bigrams or trigrams more than unigrams, at least in technical texts, this dataspace should result in another meaningful view of the data.

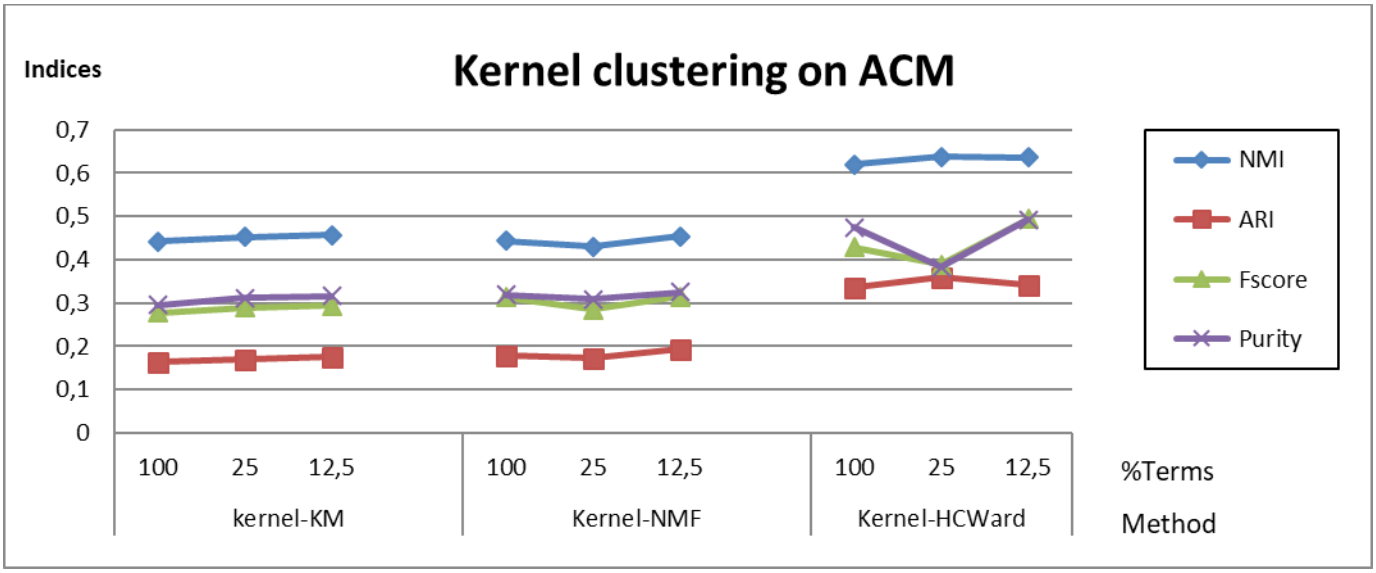

Figure 23: Kernel Clustering on ACM corpus

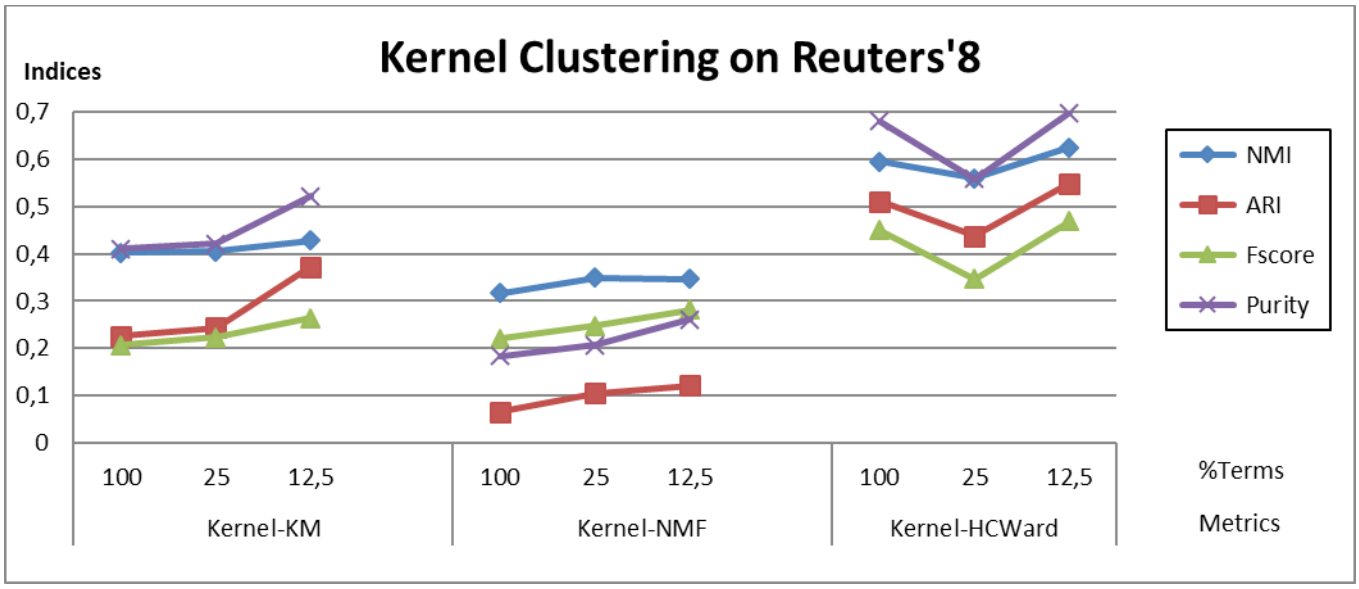

Figure 24: Kernel Clustering on Reuters'8 corpus

It is indeed the case, the surprise being that usual Kernel K-means is clearly outperformed by Ward-linkage clustering in the kernel space. We have not heard of this result elsewhere in the literature.

Given the long computation time on Ng20 corpus (3 hours), we performed only two runs with a disappointing outcome: at best, NMI=.404, ARI=.202, Fscore=.313, Purity=.336 on a 25%-truncated vocabulary.

2.7 - Graph clustering

- Louvain

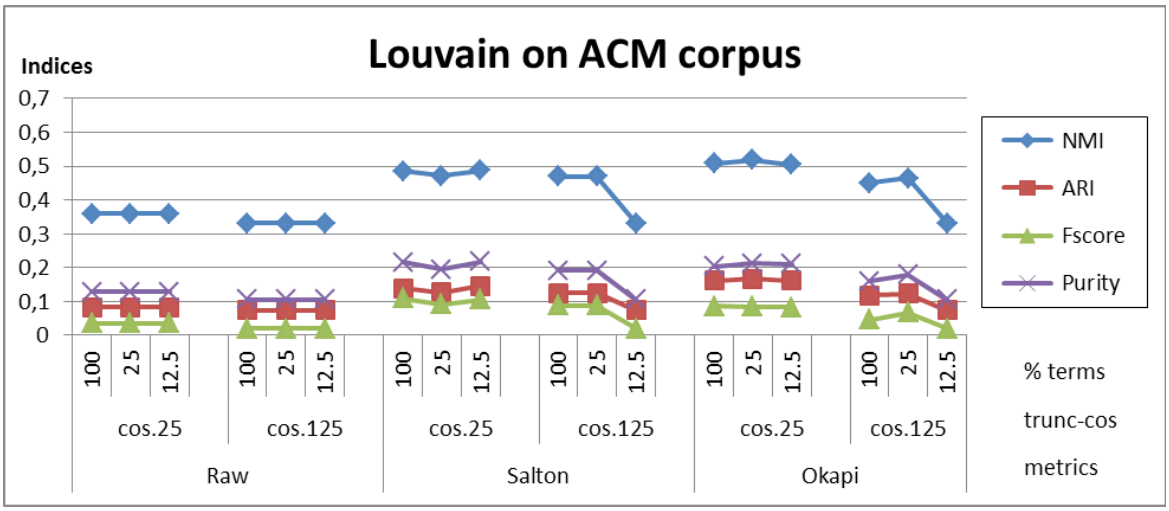

Figure 25: Louvain graph clustering on ACM corpus

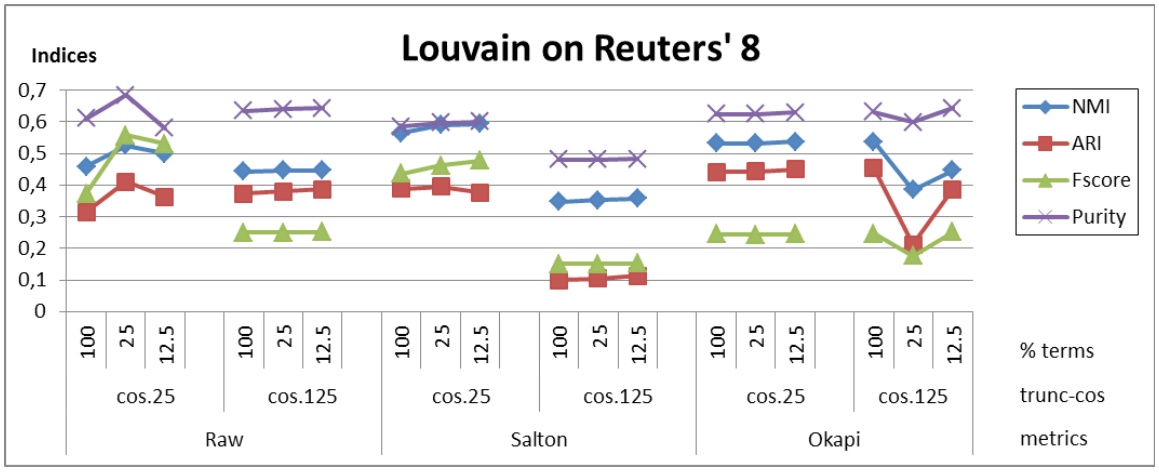

Figure 26: Louvain graph clustering on Reuters'8 corpus

The performance is not so good as using spectral HC-Ward, but shows a slight indifference to thresholds, applied to vocabulary as well as cosines. Thus a sparse "graph-like" sparse cosine matrix does not seem to be an advantage, contrary to our initial guess.

This guess led us to test a single run of Louvain (6 hours) on a doubly truncated Ng20 space (25% truncated vocabulary, 25% truncated cosines) but the performances were poor: NMI=.302, ARI=.116, Fscore=.114, Purity=.211, which seems to confirm the above statement.

- InfoMap

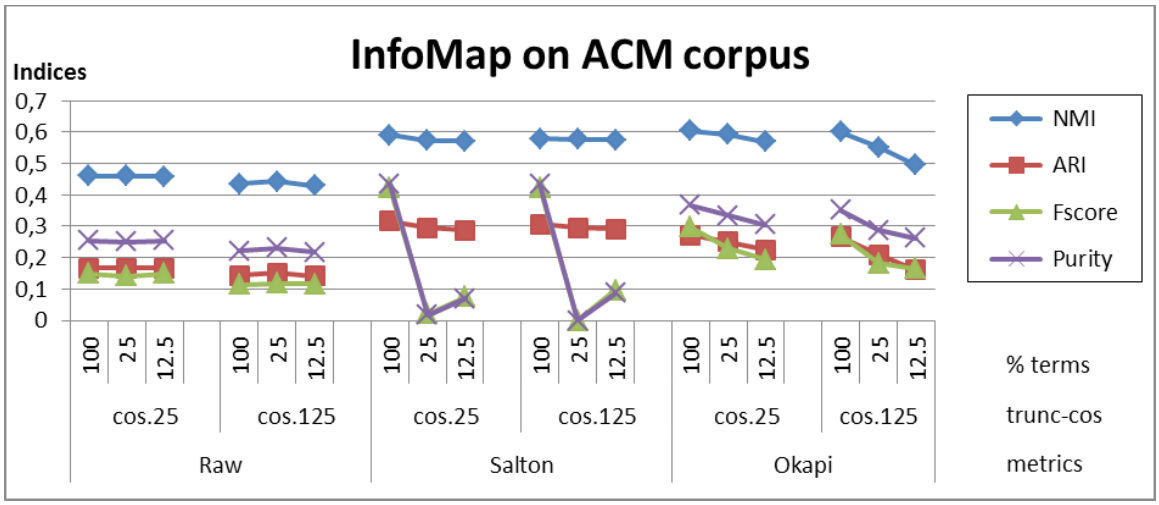

Figure 27: InfoMap graph clustering on ACM corpus

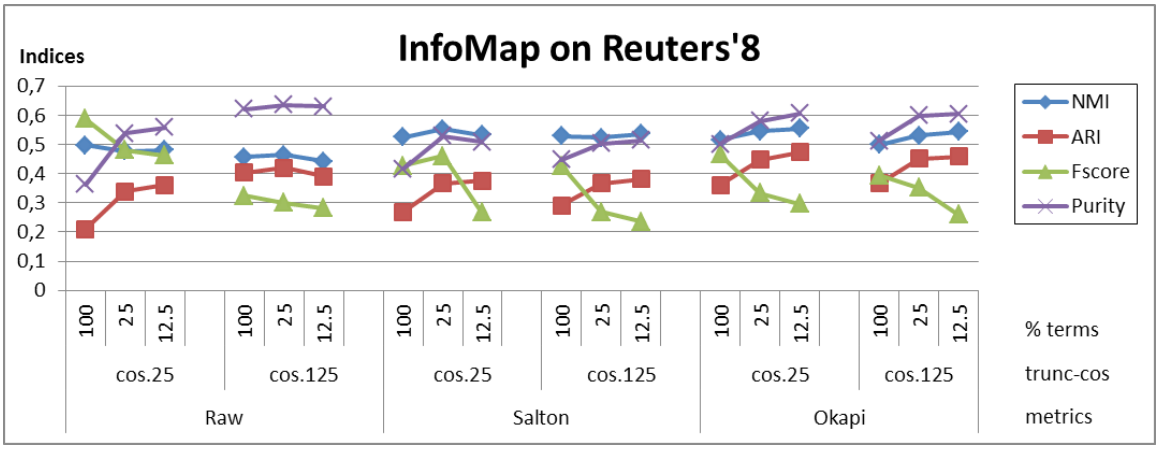

Figure 28: InfoMap graph clustering on Reuters'8 corpus

The performance is good, especially in the Okapi and Salton's dataspaces, with no sensible influence of the thresholds.

We tested two InfoMap runs on two Ng20 spaces (3 hours each), both 25% cosine-truncated, the first with 100% vocabulary, the second with 25%, resulting in deceptive scores, at best: NMI=.356, ARI=.162, Fscore=.213, Purity=.290.

3 - Synthesis

Let us focus first on the measurement tools: we can observe that in the case of the two "balanced" corpora, the four evaluation indices behave in a much parallel and orderly manner - see fig. 1, 3, 4, 6, 25. In contrast, this parallelism and regular ranking deteriorate in the Reuters'8 unbalanced corpus, and to a lesser extent when hierarchical methods are used. F-score and Purity indices may (see fig. 17, 28) or may not present (see fig. 16, 20, 21, 25) a somehow contradictory or non-monotonic behavior. A thorough investigation could perhaps explain these interesting discrepancies, but is clearly out of our present goals. We have thus chosen the more stable NMI index as a reference measure for ranking each corpus' runs (ACM: 246 runs, Re8: 237 runs, Ng20: 109 runs, summing up to 592 runs).

3 - 1 - "Top three" runs (NMI criterion):

ACM corpus:

1) Spectral HC-Ward in k*-dimension (i.e. 40) Laplacian space with 12.5%-truncated cosine measure, Okapi transformed, non-truncated vocabulary. NMI: .698, ARI: .483, Fscore: .619, Purity: .623; elapsed time: 77".

2) Spectral HC-Ward in k*-dimension (i.e. 40) Laplacian space with 37.5%-truncated cosine measure, Okapi transformed, non-truncated vocabulary. NMI: .695, ARI: .475, Fscore: .615, Purity: .618; elapsed time: 77".

3) Spectral HC-Ward in k*-dimension (i.e. 40) Laplacian space with 25%-truncated cosine measure, Okapi transformed, non-truncated vocabulary. NMI: .695, ARI: .475, Fscore: .615, Purity: .618; elapsed time: 77".

Reuters'8 corpus:

1) Kernel HC-Ward on 12.5%-truncated vocabulary, kernel raw dataspace (no Okapi, etc. transformation); NMI:.625, ARI:.547, Fscore:.451, Purity: .699; elapsed time: 767"

2) Spectral HC-average on 25%-truncated vocabulary, in 2k* (i.e. 16 dimensions) CA space with cosine measure; NMI: 622, ARI:.516, Fscore:.449, Purity: .691; elapsed time: 85073"

3) Spectral HC-average on 25%-truncated vocabulary, in k* (i.e. 8 dimensions) CA space with cosine measure; NMI:.615, ARI:.584, Fscore:.440, Purity: .724; elapsed time: 85610"

20 NewsGroups corpus:

1) NMF on non-truncated vocabulary, Okapi dataspace; NMI:.625, ARI:.548, Fscore:.469, Purity: .699; elapsed time: 107"

2) Spectral HC Ward in 2k* (i.e. 40 dimensions) CA space with cosine measure; vocabulary is not truncated; NMI:.622, ARI: .456, Fscore: .600, Purity: .624; elapsed time: 13 628"

3) K-Means on 12.5%-truncated vocabulary, Salton's dataspace; NMI:.621, ARI:.461, Fscore:.527, Purity: .594; elapsed time: 122"

These optimal runs clearly depend on the corpora. A large variety of dataspace transformations (truncated or not vocabulary, Salton's, 0kapi, or raw dataspace, kernel or Laplacian spectral space, ...) and methods (HC-Ward, K-Means, NMF) are present. It can be noted that one run only may be considered as "classic", i.e. K-Means on Salton's dataspace with a truncated vocabulary, in the sole case of Ng20, the other ones are not.

Two qualitative remarks:

- Good results on ACM corpus distinguish by the overwhelming effect of 1) Okapi weighting, 2) Hierarchical-Ward clustering method, 3) Laplacian dataspaces, 4) moderate or null vocabulary thresholding 5) k* dimensions in spectral dataspaces, 6) a weak influence of cosine thresholds, where they have to be.

- The results on Re8 and Ng20 corpora are positively impacted by a wide variety of method families (spectral first, then standard, and kernel too, to a much lesser extent), of algorithms (K-Means in the case of Ng20, HC-Ward and NMF in the case of Re8, HC-average also to a much lesser extent).

Now let us investigate what the maximum index values mean in the real world, coming back to the data. The correspondence table between classes and best cluster partition for Ng20 corpus (see figure 29) shows off a "snake" structure, more than a diagonal one: some clusters totally or partially correspond to several classes, and conversely. The #3 and #9 classes are dispersed through many clusters. The global 0.62 NMI and 0.57 Purity are not so good, after all, and this is confirmed by the 0.53 mean F-score. The reasons of the divergence between man-made categories and clusters should be thoroughly investigated in relation to experts of the application field, by examining for example the case of classes 3 and 9. This process may perhaps converge to a consensual "ground truth", or diverge, showing off the limits of the sole textual information - or the limits of a weakly linguistic term extraction, in the present case ?

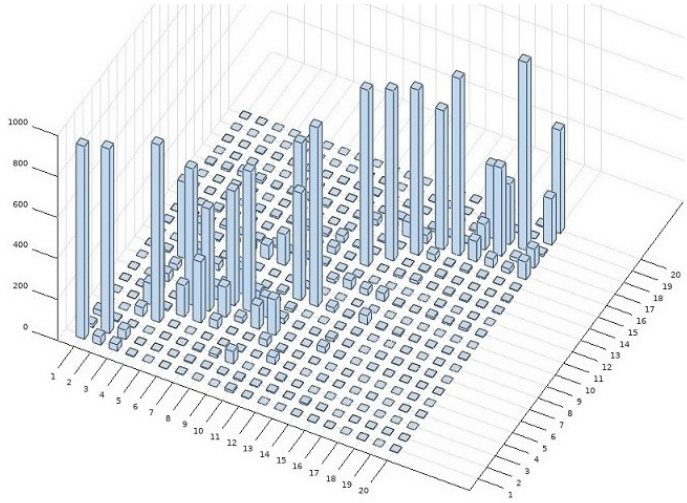

Figure 29 - Cross tabulation of clusters vs. classes on best-NMI run of 20 NewsGroups corpus. At left hand: classes; at right: clusters

To come back to figures 1 to 28, the evenness of many visually eye-catching weird observations shows that non-clearly elucidated phenomena and interactions are at work: e.g. the role that could be played by the size unevenness and number of the clusters, by the size of the vocabulary, by the type and style of texts, when matched against multiple potential classificatory points of view, by density contrasts in dataspaces, and so on. This opens up a whole unexplored research field.

Aside from that, let us search for commonalities. Examining the "Top" runs of each corpus (ranked by decreasing NMI values), a few common behaviors emerge.

- Partial commonalities: Kernel HC Ward is a clear winner in Re8 context. It ranks in the honorable 60th/246 in the ACM one, but in the “last-10” for Ng20. The combination NMF in Okapi dataspace with non-truncated vocabulary overwhelmingly dominates in Ng20 context, ranks high for ACM (in the 30th positions upon 246) but not so high for Re8 (in the 120th upon 237).

In our present state of knowledge, the variability of these results across corpora is such that it seems too early for issuing refined recommendations on the optimal use of this or that algorithm.

- Global commonalities: the best compromise we find is to combine Ward Hierarchical Clustering with cosine measures in the above-mentioned Correspondence Analysis dataspace, with a number of factors possibly exceeding k* or 2k*, which is an advantage when the "real" number of clusters is unknown, and seemingly with no decisive influence of the term threshold. In this case, the mean ratio to the best-performing run in terms of NMI is around 95%.

Another compromise is to combine Spectral K-Means with cosine measures in Correspondence Analysis dataspace. In this case, the mean ratio to the best-performing run in terms of NMI decreases to 87%. This combination is fast (12" per 20-pass run for the 19 000 Ng20 documents) and it costs O(#documents, #variables) computation time; it saves storage resources, as 25% of the original vocabulary gives the best results.

Another acceptable option is to use spectral HC-Ward in Laplacian Okapi-weighted space, which ranks first for Re8 corpus and performs correctly otherwise – its mean ratio to the best-performing run amounts to 91%.

The main problem for one to follow these recommendations is to build the spectral space(s) for real-life data. In many computer languages indeed, efficient sparse Singular Value Decomposition procedures exist, appropriate when the problem is to draw a limited number of main eigenvalues and eigenvectors from huge datatables, which is the case in the present study. Otherwise parallel graphics co-processors may be dedicated to this task.

We are not aware of any use of CA factor space for spectral K-means yet, not to mention spectral hierarchical clustering, and these original combinations will be the only cautious recommendations we could issue by now.

4 - Conclusions and perspectives

We hope we have brought some clarification to the problem of evaluating text clustering procedures, by considering separately the algorithms and the dataspaces in which they operate. We have achieved some 600 runs of a dozen algorithms and variants, in a few tens various dataspaces, on three prototypical and public access test corpora. We have brought to light an unexpected variety of optimal combinations of methods and dataspaces, from which we have derived three cautious recommendations. The variety of possible transformations and parameters requires a considerable continuation effort for improving our understanding and mastery of artificial vs. human categorization processes. We hope that this empirical survey will contribute to such an issue. In a modest first step, we will explore the influence of linguistic pre-processing: choice or elimination of word categories, comparison between taking into account multi-word expressions and kernel expansion of uniterms.

References:

[Apté et al. 1994] Apté, Chidanand, Fred Damerau, and Sholom M. Weiss. "Automated learning of decision rules for text categorization."ACM Transactions on Information Systems (TOIS)12.3 (1994): 233-251.

[Benzécri 1973] Jean-Paul Benzécri: L'analyse des correspondances, Analyse des données, Vol.2 (Dunod,Paris 1973)

[Blei et al. 2003] David M. Blei, Andrew Y. Ng, Michael I. Jordan: Latent Dirichlet Allocation, Journal of Machine Learning Research 3, 993–1022 (2003)

[Blondel et al. 2008] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, Etienne Lefebvre: Fast unfolding of communities in large networks, Journal of Statistical Mechanics: Theory and Experiment2008(10), P10008 (2008)

[Cadot et al. 2018] Martine Cadot, Alain Lelu, Michel Zitt. Benchmarking seventeen clustering methods on a text dataset. (Research Report) LORIA. 2018. ⟨hal–01532894v5⟩

[Choi et al. 2010] Choi, Seung-Seok, Sung-Hyuk Cha, and Charles C. Tappert. "A survey of binary similarity and distance measures."Journal of Systemics, Cybernetics and Informatics 8.1 (2010): 43-48.

[Cover, Thomas 1991] Cover, Thomas M., and Thomas, Joy A. "Entropy, relative entropy and mutual information."Elements of information theory2 (1991): 1-55.

[Forgy 1965] Forgy, E. (1965): Cluster analysis of multivariate data: efficiency vs. interpretability of classifications, Abstract, Biometrics, vol. 21, 768–769.

[Girolami 2002] M. Girolami. Mercer kernel-based clustering in feature space. IEEE Transactions on Neural Networks, 13(3):780–784, 2002

[Greenacre 1984] Greenacre, M.J, Theory and applications of correspondence analysis, Academic Press [1984]

[Kogan, 2007] Kogan, J. (2007). Introduction to Clustering Large and High-Dimensional Data. Cambridge University Press.

[LABIC data]http://sites.labic.icmc.usp.br/text_collections/; checked on 8/4/2019

[LABIC stemmer] http://sites.labic.icmc.usp.br/tpt/; checked on 8/4/2019

[Lebart et al. 1998] L. Lebart, A. Salem, L. Berry, Exploring Textual Data, Kluwer Academic Publisher, Dordrecht, Boston, 1998

[Lee, Seung 1999] Lee, Daniel D., and H. Sebastian Seung. "Learning the parts of objects by non-negative matrix factorization."Nature401.6755 (1999): 788-791.

[Legendre, Gallagher 2001] Legendre, P. & Gallagher, E.D, Ecologically meaningful transformations for ordination of species data - Oecologia (2001) 129: 271. https://doi.org/10.1007/s004420100716

[Lewis et al. 2004] Lewis, D. D., Yang, Y., Rose, T. G., & Li, F. (2004). Rcv1: A new benchmark collection for text categorization research. Journal of machine learning research,5(Apr), 361-397.http://www.daviddlewis.com/resources/testcollections/rcv1/

[Mc Queen 1967] Mac Queen, J. (1967): Some methods for Classification and Analysis of Multivariate Observations. In: Proc. 5th Berkeley Symp. Math. Stat. Proba., 281–297.

[Meila, Shi 2000] Marina Meila, Jianbo Shi: Learning Segmentation by Random Walks, NIPS'00: Proceedings of the Neural Information Processing Systems Conference, Denver, CO 2000, ed. by Todd K.Leen, Thomas G. Dietterich, Volker Tresp (MIT

[Müllner 2013] Daniel Müllner, fastcluster: Fast Hierarchical, Agglomerative Clustering Routines for R and Python, Journal of Statistical Software 53 (2013), no. 9, 1–18,

[Murtagh, Legendre 2014] Fionn Murtagh, Pierre Legendre - Ward's Hierarchical Clustering Method: Clustering Criterion and Agglomerative Algorithm Journal of Classification, 31 (3), 274-295, 2014

[Murtagh 1983] Fionn Murtagh. 1983. A survey of recent advances in hierarchical clustering algorithms. The Computer Journal. Volume 26, Number 4, pp. 354-359

[Murtagh 1984] F. Murtagh. 1984. Complexities of Hierarchic Clustering Algorithms: the state of the art. Computational Statistics Quarterly, 1: 101–113.

[Rand 1971] Rand, William M. "Objective criteria for the evaluation of clustering methods."Journal of the American Statistical association 66.336 (1971): 846-850

[Rossi et al. 2013] Rossi, Marcacini, Oliveira, Rezende : Benchmarking Text Collections for Classication and Clustering Tasks No 395 ICMC TECHNICAL REPORT. São Carlos, SP, Brazil 2013

[Rosvall, Bergstrom 2007] Martin Rosvall, Carl T. Bergstrom: An information-theoretic framework for resolving community structure in complex networks, Proceedings of the National Academy of Sciences104(18), 7327–7331 (2007)

[Van Mechelen et al. 2018] Iven Van Mechelen, Anne-Laure Boulesteix, Rainer Dangl, Nema Dean, Isabelle Guyon, Christian Hennig, Friedrich Leisch, Douglas Steinley - Benchmarking in cluster analysis: A white paper. arXiv:1809.10496v2 [stat.OT]

[Van Rijsbergen 1979] Van Rijsbergen, C. J. (1979). Information Retrieval (2nd ed.). Butterworth-Heinemann.

[Von Luxburg 2007] Von Luxburg, Ulrike. "A tutorial on spectral clustering."Statistics andcomputing17.4 (2007): 395-416.

[Ward 1963] Ward, Joe H. (1963). "Hierarchical Grouping to Optimize an Objective Function". Journal of the American Statistical Association. 58 (301): 236–244

[Zitt et al. 2019] Michel Zitt, Alain Lelu, Martine Cadot, Guillaume Cabanac. "Bibliometric delineation of scientific fields" inWolfgang Glänzel; Henk F. Moed; Ulrich Schmoch; Michael Thelwall. *Handbook of Science and Technology Indicators*, Springer International Publishing, 2019, 978-3-030-02510-6.