GENETIC LOCALIZATION OF SEMANTIC PROTOTYPES FOR MULTICLASS RETRIEVAL OF DOCUMENTS

Daniel Devatman Hromada[123]

[1] *Institute of Robotics and Cybernetics,*
Faculty of Electrical Engineering and Information Technology,
Slovak University of Technology in Bratislava,
Ilkovičova 3, 812 19 Bratislava, Slovak Republic
(e-mail: hromi at giver dot eu)

[2] *Laboratoire ChART (Cognition Humaine et Artificielle)*
Université Paris 8,
2, rue de la Liberté
93526 St Denis Cedex 02

[2] *Laboratory of Computational Art*
Faculty of Design
Berlin University of Arts
Grunewaldstrasse 2-5
10823 Berlin-Schöneberg

Abstract: *The paper presents a novel method of multiclass classification. The method combines the notions of dimensionality reduction and binarization with notions of category prototype and evolutionary optimization. It introduces a supervised machine learning algorithm which first projects documents of the training corpus into low-dimensional binary space and subsequently uses canonical genetic algorithm in order to find a constellation of prototypes with highest classificatory pertinence. Fitness function is based on a cognitively plausible notion that a good prototype of a category C should be as close as possible to members of C and as far as possible to members associated to other categories. In case of classification of documents contained in a 20-newsgroup corpus into 20 classes, our algorithm seems to yield better results than a comparable deep learning "semantic hashing" method which also projects the semantic data into 128-dimensional binary (i.e. 16-byte) vector space.*

Keywords: multiclass classification, dimensionality reduction, evolutionary computing, prototype theory of categorization, light stochastic binarization, canonic genetic algorithm, supervised machine learning

1 INTRODUCTION

In computational theories and models learning, one generally works with two types of models: regression and classification. While in regression models one maps continuous input domain onto continuous output range, in models of classification, one aims to find mappings able to project input objects onto a finite set of discrete output categories.

This article introduces a novel means of construction of a particular type of the latter kind of learning models. Due to finite and discrete nature of its output range, classification - also called categorization by more cognition-oriented researchers - seems to be of utmost importance in any cognitively plausible (Hromada 2014*a*) model of learning. But under these terms, two distinct meanings are confounded and the term categorization thus often represents both:

1. process of learning (e.g. inducing) of categories
2. process of retrieving information from already learned (induced) categories

which crudely correspond to training, resp. testing phases of supervised learning algorithms.

Rest of this section, as well as section 2, shall more closely introduce an approach combining notions of category prototype, dimensionality reduction and evolutionary computing in order to yield a potentially "cognitively plausible" means of supervised machine learning of a multiclass classifier. Sections 3 shall present specificities of a Natural Language Processing (NLP) simulation which was executed in order to assess the feasibility of the algorithm and in section 4, obtained results shall be compared with comparable "deep learning" semantic hashing technique of (Salakhutdinov & Hinton 2009). The article shall be concluded, in section 5, with few remarks aiming to integrate whole research into more generic theories of neural and universal darwinism.

1.1 Geometrization of Categories

In contemporary cognitive science, categories are often understood as entities embedded in an Δ-dimensional feature space (Gärdenfors 2004). The most fundamental advantage of such models, whose computer sciences counterparts are so-called "vector symbolic architectures" (VSAs) (Widdows & Cohen 2014), is their ability to geometrize one's data, i.e. to represent one's dataset in a form which allows to measure distances (similarities) between individual items of the dataset.

Thus, even entities like "word meanings" or "concepts" can be geometrically represented, either as points, vectors or subspaces of the envelopping vector space S. One can subsequently measure distances between such representations, e.g. distance of the meaning of the word "dog" from the meaning of "wolf" or "cat" etc. Geometrization of one's dataset once effectuated, space S can be subsequently partitioned into a set R of $|C|$ regions $R = R_1, R_2, ..., R_{|C|}$. In unsupervised scenario, such partitioning is often done by means of diverse clustering algorithms, the most canonic among which being the k-means algorithm (MacQueen et al. 1967). Such clustering mechanisms often characterize candidate cluster C_X in terms of a geometric centroid of the members of the cluster. Feasibility of a certain partition is subsequently assessed in terms of "internal clustering criteria" which often take into account distances among such centroids.

In the rest of this article, however, we shall aim to computationally implement a supervised learning scenario and instead of working with the notion of category's geometric centroid, our algorithm shall be based upon the notion of category's prototype. The notion of the prototype was introduced into science notably by theory of categorization of Eleanore Rosch which departed from the theoretical postulate that:

"*the task of category systems is to provide maximum information with the least cognitive effort*" (Rosch 1999)

In seminal psychologic and anthropologic studies which have followed, Rosch have realized that people often characterize categories in terms of one of their most salient members. Thus, a prototype of category C_X can be most trivially understood as such a member of C_X which is the most promiment, salient member of C_X. For example "apples" are prototypes of category "fruit" and "roses" are prototypes of category "flowers" in western cultural context.

But studies of Rosch had also suggested another, more mathematic, notion of how prototypes can be formalized and represented. A notion which is based upon the notion of closeness (e.g. "distance") in a certain metric space:

"*items rated more prototypical of the category were more closely related to other members of the category and less closely related to members of other categories than were items rated less prototypical of a category*" (Rosch & Mervis 1975)

Given that this notion is essentially geometric, the problem of discovery of a set of prototypes can be potentially operationalized as a problem of minimization of a certain fitness function. The fitness function, as well as means how it can be optimized, shall be furnished in section 2. But before doing so, let's first introduce certain computational tricks which allow to reduce the computational cost of such search of the most optimal constellation of prototypes.

1.2 Radical Dimensionality Reduction

There is potentially an infinite number of ways how a dataset D consisting of |D| documents can be geometrized into a $\Delta-$dimensional space S. In NLP, for example, one often looks for occurences of diverse words in the documents of the dataset (e.g. corpus). Given that there are |W| distinct words occuring in |N| documents of the corpus, one used to geometrize the corpus by means of a N * M co-occurrence matrix M whose X-th row vector represents the X-th document N_X, Y-th column vector represents the Y-th word W_Y and the element on position $M_{X,Y}$ represents the number of times W_Y occured in N_X.

Given the sparsity of such co-occurrence matrices as well as for other reasons, such bag-of-words models are more or less abandoned in contemporary NLP practice for sake of more dense representations, whereby the dimensionality of the resulting space, d, is much less than $|W|$, $d \ll |W|$. Renowned methods like Latent Semantic Analysis (LSA) (Landauer & Dumais 1997) set aside because of their computational cost, we shall use the Light Stochastic Binarization (LSB) (Hromada 2014*b*) algorithm to perform the most radical dimensionalityreducing geometrization possible.

LSB is an algorithm issued from the family of algorithms based on so-called random projection (RP). Validity and feasibility of all these algorithms, be it Random Indexing (RI, (Sahlgren 2005)) or Reflective Random Indexing (RRI,(Cohen et al. 2010)) is theoretically founded on a so-called lemma of Johnson-Lindenstrauss, whose corollary states that "*if we project points in a vector space into a randomly selected subspace of sufficiently high dimensionality, the distances between the points are approximately preserved*" (Sahlgren 2005).

Methods of application of this lemma in concrete NLP scenarios being described in references above, we precise that LSB can be labeled as "most radical" variant of RP-based algorithms because:

- it tends to construct spaces with as small dimensionality as possible (in LSB, $d < 300$; in RI or RRI models, $d > 300$)

- LSB tends to project the data onto binary and not real or complex spaces

It can be, of course, the case that such dimensionality-reduction and binarization can lead to certain decrease of discriminative accuracy of LSB-produced spaces. On the other hand, given that dimensionality reduction and binarization neccessary bring about reduction of computational complexity of any subsequent algorithm which could be used to explore the resulting space S, such decrease of accuracy is to be more swiftly counteracted by subsequent optimization. The goal of this study is to explore whether such *post hoc* optimization of classifiers operating within dense, binary, LSB-produced spaces is possible, and whether the combination of the two can be used as a novel means of machine learning.

But before describing in more closer such evolutionary optimizations, let's precise that because of its low-dimensional and binary nature, LSB can also be understood as yielding a sort of "hashing function" aiming to attribute similar hashes to similar documents and different hashes to different documents. In this sense, LSB is similar to approaches like Locality Sensitive Hashing (LSH, Datar et al. (2004)) or Semantic Hashing (SH, Salakhutdinov & Hinton (2009)) often used, or at least presented, as *the* solution of multiclass classification of BigData corpora. It is with the results of the latter, "deep-learning" approach, that we shall compare our own results in section 4.

2 GENETIC LOCALIZATION OF SEMANTIC PROTOTYPES

Let $D = \{d_1, ..., d_{|D|}\}$ be a training dataset consisting of $|D|$ documents to which the training dataset attributes one among $|L|$ corresponding members of set of class labels $L = \{L_1, ..., L_{|L|}\}$.

Let Γ denote a set of classes $\Gamma = C_1, ..., C_|L|$ whose individual members are also sets containing indices of members of D to which a same label L_l is attributed in the training corpus (e.g. $C_1 = \{1, 2, 3\}$ if training corpus attributes label 1 only to d_1, d_2 and d_3).

Let $H = \{h_1, ..., h_{|D|}\}$ be a set of Δ-dimensional binary vectors attributed to members of D by a hashing function F_H, i.e. $h_X = F_H(d_X)$.

Let S be a Δ-dimensional binary (Hamming) space into which members of H were projected by application of mapping F_H.

Then a classificatory pertinence F_{CP} of the candidate prototype P_K of K-th class ($K \leq |C|$) can be calculated as follows:

$$F_{CP}(P_K) = \alpha \sum_{t \in C_K} F_{hd}(h_t, P) - \omega \sum_{f \not\subset C_K} F_{hd}(h_f, P) \quad (1)$$

whereby P denotes the position of the prototype in S, F_{hd} denotes the Hamming distance[1], h_t denotes the hash "true" document belonging to same class as the prototype, h_f is the vector of the "false" document belonging to some other class of the training corpus and α and ω are weighting parameters.

In simpler terms, an ideal prototype of category C is as close as possible to members of C and as far away as possible from members of other categories.

Given such a definition of an ideal prototype, an ideal $|C|$-class classifier I can be trained by searching for such a set $P = \{P_1, ..., P_{|L|}\}$ of individual prototypes, which minimize their overall classification pertinence:

$$I = min \sum_{K=0}^{K=|L|} F_{CP}(P_K) \quad (2)$$

In simpler terms, an ideal $|C|$-class classifier I is composed of $|C|$ individual prototypes which are as close as possible to documents of their respective categories, and as far away as possible from all other documents.

Equations 1 and 2 taken together, one obtains a fitness function which can be optimized by evolutionary computing algorithms. And given that one explores the prototypical constellations embedded in a binary space, one can use canonical genetic algorithms (CGAs, Goldberg (1990)) for the optimization of the problem of

[1] Hamming distance of two binary vectors h_1 and h_2 is the smallest number of bits of h_1 which one has to flip in order to obtain h_2. It is equivalent to a number of non-zero bits in a $XOR(h_1, h_2)$ binary vector.

discovery of ideal constellation of most pertinent prototypes. We choose CGAs for three principal reasons:

Primo, we choose CGAs mainly for their property, proven in Rudolph (1994), to converge to global optimum in finite time if ever they are endowed with the best-individual protecting, elitist strategy. Secundo, one can obtain practically useful and exploitable increase in speed simply due to the fact that CGAs are conceived to process binary vectors and do so on CPUs which are essentially built for processing such vectors. Tertio, CGAs offer a canonical, well-defined, "baseline" gateway to much more sophisticated evolutionary computing (EC) techniques and are well understood by both neophytes as well as the most experts of the EC community.

For this reason, we consider as superflous to describe in closer detail the inner workings of a CGA: instead, references (Goldberg 1990, Rudolph 1994) are to be followed and read. Given that the particular values of mutation and cross-over parameters shall be specified in the following section, the only thing which in which the reader now needs to be reassured is her correct understanding of the nature of datas structures which the algorithm hereby proposed shall implement, in order to encode an individual $|C|$-class classifier:

Given that equation 1 defines a prototype candidate as a position in Δ-dimensional Hamming space and given that equation 2 stipulates that an ideal $|C|$-class classifier is to be composed of representations of $|C|$ ideal prototype candidates, the data structure representing an individual solution can be constructed by a simple concatenation of $|C|$ Δ-dimensional vectors. Thus, the individual members of the populations which the CGA shall optimize are, *in essentia*, nothing else than binary strings of length $|C|*\Delta$.

3 CORPUS AND TRAINING PARAMETERS

In order to be able to compare the performance of our algorithm with non-optimized LSB and SH, same corpus and dimensionality parameters were chosen as those, which are already reported in the previous studies (Salakhutdinov & Hinton 2009, Hromada 2014*b*). Thus, dimensionality of the resulting binary hashes was Δ=128. Every document of the corpus was hence attributed a 16-byte long hash.

A so-called "20newsgroups" corpus[2] has been used. The corpus contains 18,845 postings taken from the Usenet newsgroup collection divided into training set containg 11,314 postings, 7531 being the testing set ($|D_{training}| = 11313, |D_{testing}| = 7531$). Both training and testing subsets are divided into 20 different newsgroups which correspond each to a distinct topic. Given that every distinct topic represents a distinct category label, $|C| = 20$.

Documents of the corpus were subjected to a very trivial form of pre-processing: documents were split into word-tokens by means of [$\hat{\backslash}$w] separator. Stopwords contained in PERL library Lingua::StopWords were subsequently discarded. 3000 word types with highest "inverse document frequency" value were used as initial terms to which the initial random indexing iteration attributed 4 non-zero values. Hashing function $F_H = LSB(\Delta = 128, Seed = 3, Iterations = 2)$ because there were 2 "reflective" iterations preceding the ultimate stage of "binarization".

Once hashes were attributed to all documents of the corpus, the Hamming space S was considered as constructed and stayed unmodified during all phases of subsequent optimizations and evaluations. As CGA-compliant algorithm, the optimization applied generated the new generation by crossing over two parent solutions chosen by the fitness proportionate (e.g. roulette wheel) selection operator. Each among 2560 (128*20) genes was subsequently mutated (i.e. a correspondent bit was flipped to its opposite value) with probability of 0.1%. Population contain 200 individuals, zeroth generation was randomly generated. Elitist strategy was implemented so that all individuals with equally best fitness survived intact the transition to future generation. Parameters α and ω (e.g. equation 1) used in fitness estimation were both set to 1.

Information concerning the category labels guided the optimization during the training phase. During the testing phase, such information was used only for evaluation purposes. Multiple independent runs were executed and values of precision and recall were averaged among the runs in order to reduce the impact of stochastic factors upon the final results.

4 EVALUATION AND RESULTS

Every 250th generation, classificatory accuracy of an individual solution with minimal overall classification pertinence (c.f. equation 2) was evaluated in regards to 7531 documents contained in the testing part of the corpus. Following aspects of classifier's performance were evaluated in order to allow comparison with the results with Precision-Recall curves presented in (Salakhutdinov & Hinton 2009, Hromada 2014*b*):

$$Precision = \frac{\text{Number of retrieved relevant documents}}{\text{Total number of retrieved documents}}$$

[2] http://qwone.com/ jason/20Newsgroups/

$$Recall = \frac{\text{Number of retrieved relevant documents}}{|D_{testing}|}$$

The notion of relevancy is straightforward: an arbitrary document D_T contained in the testing corpus is considered to be relevant to query document D_Q if and only if they were both labeled with the same category label, $L_Q = L_T$.

On the other hand, the correct understanding of what is meant by "retrieved" is the key to correct understanding of the core idea behind the functionality of the algorithm hereby proposed. That is: **the prototypes induced by the CGA optimization are to be used as retrieval filters**.

We precise: given a hash h_Q of a query document d_Q, one can easily identify - among $|C|$ prototypes encoded as components of an quasi-ideal constellation I furnished by the CGA - such a prototype P_N which is nearest to h_Q. Subsequently, each among N documents whose hashes are N nearest neighbors of the prototype P_N, should be considered as retrieved by d_Q. Prototypes discovered during the training phase therefore primarily specify, during the testing phase, which documents are to be considered as retrieved, and which not. For all LSB curves present on Figure 1, the size of such retrieval neighborhood was set to N=2000.

Also, in order to obtain viable precision-recall curves, Radius R=(0, ..., Δ = 128) of the Hamming ball was used as a tradeoff parameter. For every datapoint of the plot on Fig. 1, h_N was considered as retrieved by query h_Q only if the hamming distance of query and the candidate document was smaller than R ($hd(h_Q, h_N) > R$). Points on the very left of the plot correspond thus correspond to R=0 (i.e. h_Q and h_N collide), while points on the right correspond to R=128 (i.e. h_Q does not have a single bit in common with h_N).

As comparison of curves on the figure indicates, *biggest increase in performance is attained by decision to use prototypes as retrieval filters*. Thus, when one uses the most fit among 200 randomly chosen prototype constellations as a retrieval filter (c.f. curve CGA1(LSB)), one obtains significantly better results than when does not use any prototypes at all (c.f. curve "Plain LSB"). If the process is followed by further genetic optimization (c.f. CGA500 for situation after 500 generations), one observes a non-negligeable increase of precision in the high recall region of the spectrum. But it can also be seen that the optimization has its limits, hence there is a slight decrease between 500th and 1000th generation which potentially corresponds to situation whereby the induced prototype constellation tends to overfit the training dataset. This leads to subsequent decrease in overall accuracy of classification of documents contained in the testing dataset.

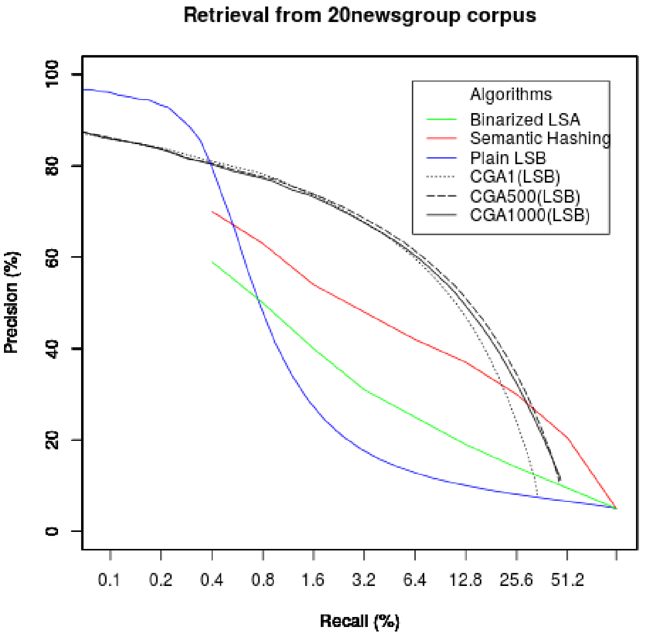

Fig. 1: Retrieval and 20-class classification performance in 128-dimensional binary spaces. Non-LSB results are reproduced from Figure 6 of study (Salakhutdinov & Hinton 2009), plain LSB from (Hromada 2014*b*).

Figure 1. also suggests that the genetic discovery of sets of prototypes - and their corresponding use as retrieval filters - seems to produce results which are better[3] than those produced by both binarized Latent Semantic Analysis or SH.

5 CONCLUSION

Results hereby presented indicate that supervised localisation of constellations of semantic prototypes can significantly increase accuracy of classifiers which use such constellations as retrieval filters.

Given that the localization of such constellations is governed by the training corpus but the increase is also significant in case when one confronts the system with previously unseen testing corpus, we are allowed to state that **the algorithm hereby introduce is capable**

[3] Exception to this is SH's 20% precision at recall level of 51.2%. Note, however, that since on page 6 of their article, Salakhutdinov & Hinton (2009) claim to have used their hashes as retrieval filters of neighborhood of size N=100, and given that the every size of the category in a 20newsgroup corpus ≈ 390 documents, such a result is not even theoretically possible. This is so because even in case the classifying system would retrieve only the relevant documents (i.e. precision would be 100%) the maximal attainable recall would still be just 100/390 ≈ 25.6%. Both authors were contacted by mail with a request to rectify possible misunderstanding. Unfortunately, none of them replied.

of generalization. This was principially attained by combination of following ideas:

1. projection of documents into low-dimensional binary space
2. definition of fitness of prototype in terms of distances to both documents of its category, as well as distance to document of other categories
3. search for fittest prototype constellations
4. use of the most fit prototype constellation as a sort of retrieval filter

In spite of its generalizing and thus "machine learning" capabilities, our algorithms is essentially a non-connectionist one. Thus, instead of introducing synapses between neurons, or speaking about edges between nodes of the graph, briefly, instead of speaking about *deep learning of multi-layer encoders of stacks of Restricted Boltzmann Machines fine-tuned by back-propagation* as (Salakhutdinov & Hinton 2009) do - we have found as more preferable to reason in geometric and evolutionary terms. It is indeed due to this "geometric" perspective that the computational complexity of the algorithm is fairly low: $\Delta|D||C|$ for evaluation of fitness of one individual prototype constellation. [4]

In practical terms, it is also advantageous that both fitness function evaluation as well as final retrieval assess distances in terms of binary hamming distance measure. In both cases, one can use basic logical operations like XOR + some basic assembler instructions which would furnish indices allowing to execute sort of "conceptual goniometry" with particular swift and ease. Given these properties + the fact that hashes which are manipulated are fairly small (in one gigabyte of memory, one can store hashes for 8 million documents), one can easily predict existence of future application-specific integrated circuit (ASIC) potentially executing billions query2document comparisons per second.

Computational aspects aside, our primary motive in developing the algorithm hereby proposed was to furnish a sort of cognitively plausible (Hromada 2014*a*) "experimental proof" for our doctoral Thesis which postulates that a sort of evolutionary process exists not only in the realm of biological species, but also in realms populated by "species" of a completely different kind. Id est, in realms of linguistic structures and categories, in realms of word meanings, concepts and, who knows, maybe even in the realm of mind itself.

Being uncertain about whether the results hereby presented demonstrate, with sufficient clarity, that it is reasonable to postulate not only neural (Edelman 1987), but also intramental evolutionary processes, we conclude by saying that the formula hereby introduced offers a simple yet quite effective means of solving the problem of multiclass categorization of texts.

REFERENCES

Cohen, T., Schvaneveldt, R. & Widdows, D. (2010), ‘Reflective random indexing and indirect inference: A scalable method for discovery of implicit connections’, *Journal of Biomedical Informatics* **43**(2), 240–256.

Datar, M., Immorlica, N., Indyk, P. & Mirrokni, V. S. (2004), Locality-sensitive hashing scheme based on p-stable distributions, *in* ‘Proceedings of the twentieth annual symposium on Computational geometry’, ACM, pp. 253–262.

Edelman, G. M. (1987), *Neural Darwinism: The theory of neuronal group selection.*, Basic Books.

Gärdenfors, P. (2004), *Conceptual spaces: The geometry of thought*, MIT press.

Goldberg, D. E. (1990), ‘Genetic algorithms in search, optimization & machine learning’, *Addison-Wesley* .

Hromada, D. D. (2014*a*), Conditions for cognitive plausibility of computational models of category induction, *in* ‘Information Processing and Management of Uncertainty in Knowledge-Based Systems’, Springer, pp. 93–105.

Hromada, D. D. (2014*b*), Empiric introduction to light stochastic binarization, *in* ‘Text, Speech and Dialogue’, Springer, pp. 37–45.

Landauer, T. K. & Dumais, S. T. (1997), ‘A solution to plato’s problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge.’, *Psychological review* **104**(2), 211.

MacQueen, J. et al. (1967), Some methods for classification and analysis of multivariate observations, *in* ‘Proceedings of the fifth Berkeley symposium on mathematical statistics and probability’, Vol. 1, California, USA, pp. 281–297.

Rosch, E. (1999), ‘Principles of categorization’, *Concepts: core readings* pp. 189–206.

Rosch, E. & Mervis, C. B. (1975), ‘Family resemblances: Studies in the internal structure of categories’, *Cognitive psychology* **7**(4), 573–605.

Rudolph, G. (1994), ‘Convergence analysis of canonical genetic algorithms’, *Neural Networks, IEEE Transactions on* **5**(1), 96–101.

Sahlgren, M. (2005), An introduction to random indexing, *in* ‘Methods and applications of semantic indexing workshop at the 7th international conference on terminology and knowledge engineering, TKE’, Vol. 5.

Salakhutdinov, R. & Hinton, G. (2009), ‘Semantic hashing’, *International Journal of Approximate Reasoning* **50**(7), 969–978.

Widdows, D. & Cohen, T. (2014), ‘Reasoning with vectors: a continuous model for fast robust inference’, *Logic Journal of IGPL* p. jzu028.

[4] In future study, we aim to explore the performance of slightly modified fitness function whose complexity $\Delta|D| + |C|^2$ could be of particular interest in cases of huge datasets (i.e. big |D|) with fairly limited number of classes (|C|).