Revisiting Fisher Kernels for Document Similarities

Martin Nyffenegger¹, Jean-Cédric Chappelier¹, and Éric Gaussier²

¹ Ecole Polytechnique Fédérale de Lausanne, Switzerland

² Xerox Research Center Europe, Meylan, France

Abstract. This paper presents a new metric to compute similarities between textual documents, based on the Fisher information kernel as proposed by T. Hofmann. By considering a new point-of-view on the embedding vector space and proposing a more appropriate way of handling the Fisher information matrix, we derive a new form of the kernel that yields significant improvements on an information retrieval task. We apply our approach to two different models: Naive Bayes and PLSI.

1 Introduction

This paper presents a new way of computing similarities between textual documents based on Fisher kernels and inspired from the method introduced by Hofmann for Probabilistic Latent Semantic Indexing (PLSI) [1]. Fisher kernels define similarities between probabilistic models, based on information-geometry considerations [2]. When applied to documents, they allow the underlying semantics of the documents being compared to be taken into account. They can furthermore be used in both supervised and unsupervised learning contexts.

The next section of this paper introduces the Fisher kernel and the probabilistic models we consider for documents. We then propose two improvements: one arising from considerations about the underlying vector spaces in Sect. 3, and one from the Fisher information matrix approximation in Sect. 4. Finally, the new formulas we derive are evaluated on several Information Retrieval (IR) test collections in Sect. 5.

2 Fisher Kernels for Document Models

Let $\mathcal{D}$ be a collection of N documents: $\mathcal{D} = \{d_1, \dots, d_N\}$. Each document consists of a bag-of-words taken from a finite vocabulary $\mathcal{W}$ of M words: $\mathcal{W} = \{w_1, \dots, w_M\}$. Latent class models rely on unobserved “semantic classes”, represented by some unobserved class variables out of $\mathcal{Z} = \{z_1, \dots, z_K\}$. A class represents a set of words describing the same (or a few related) concept(s) or topic(s). We here focus on two latent class models: Naive Bayes (also called “mixture of unigrams”) [3,4] and PLSI [5,1], but our method can directly be applied to other latent class models, and more generally to mixture models, for textual documents.

In the Naive Bayes model, each document is generated from a mixture of multinomials: $p(d) = \sum_{z \in \mathcal{Z}} p(d|z) p(z)$, with $p(d|z) = \prod_{w \in \mathcal{W}} p(w|z)^{n(d,w)}$, where $n(d, w)$ is the number of occurrences of word w in document d . The parameters of this model are $\theta^{\text{NB}} = \left(p(z_k), p(w_j|z_k) \right)$, for $k = 1 \dots K$ and $j = 1 \dots M$.¹

In the PLSI model, a collection of documents is modelled as a bag of co-occurring (document, word) pairs, the log-likelihood of the collection being $l_d^{\text{PLSI}}(\theta) = \sum_{d \in \mathcal{D}} \sum_{w \in \mathcal{W}} n(d, w) \log p(d, w)$. In this model, documents and words are assumed to be independent conditionally on latent classes, thus: $p(d, w) = \sum_{z \in \mathcal{Z}} p(z) p(d|z) p(w|z) = p(d) \sum_{z \in \mathcal{Z}} p(w|z) p(z|d)$. The parameters of this model are $\theta^{\text{PLSI}} = \left(p(z_k), p(w_j|z_k), p(d_i|z_k) \right)$, for $k \leq K$, $j \leq M$, and $i \leq N$.¹

An important difference between Naive Bayes and PLSI lies in the fact that the latter is not, strictly speaking, a generative model: Its reliance on the parameters $p(d_i|z_k)$ prevents the generation of new documents (for which no d_i exist). Yet, this model has proved useful in many practical situations, and has been successfully used in different fields [7,8].

The Fisher kernel, first introduced by Jaakkola and Haussler [2], constitutes a similarity function between data points, derived from a probabilistic model of the data. It measures to which extent two data points “stretch” the model in the same direction: $\mathcal{K}\langle d, d' \rangle = (\nabla_{\theta} l_d(\theta))^T G(\theta)^{-1} \nabla_{\theta} l_{d'}(\theta)$, where $l_d(\theta)$ is the log-likelihood of the parameters θ for document d , and $G(\theta)$ is the “Fisher information matrix”, defined as the covariance matrix of the Fisher score.

The Fisher information matrix plays an important role in the above definition as it makes the kernel independent of the chosen parameterization (for equivalent parameterizations, i.e. related through diffeomorphisms). As the Fisher information matrix is difficult to compute, and as its form changes according to the parameterization adopted, it is natural to use a parameterization in which the information matrix can be approximated by the identity matrix. For both Naive Bayes and PLSI, the “square-root re-parameterization” has been deemed to play this role [2,1]. We also use this parameterization here.

For Naive Bayes, $l_d^{\text{NB}}(\theta) = \log \sum_{z \in \mathcal{Z}} p(z) \prod_{w \in \mathcal{W}} p(w|z)^{n(d,w)}$, and the associated Fisher kernel is²:

$$\mathcal{K}^{\text{NB}}\langle d, d' \rangle = \sum_{z \in \mathcal{Z}} \frac{p(z|d)p(z|d')}{p(z)} + \sum_{w \in \mathcal{W}} n(d, w)n(d', w) \sum_{z \in \mathcal{Z}} \frac{p(z|d)p(z|d')}{p(w|z)}. \quad (1)$$

For PLSI, $l_d(\theta)$ amounts to:

$$l_d^{\text{PLSI}}(\theta) = \sum_{w \in \mathcal{W}} n(d, w) \log \sum_{z \in \mathcal{Z}} p(z) p(w|z) p(d|z).$$

¹ These parameters are in fact related through normalization constraints. However, as already pointed out in [1], and confirmed by our own experiments in [6], these constraints can experimentally be ignored in the kernel derivations and have thus, for the sake of clarity, not been included here.

² We here omit the derivation, which is mainly technical and similar to the one in [9], even though from a different document likelihood.

Introducing $\hat{p}(d, w) = n(d, w)/W$, where $W = \sum_d \sum_w n(d, w)$ is the total number of words in the collection, and making the assumption $\sum_w \frac{\hat{p}(w, d)}{p(w, d)} p(w|z) \approx 1$, the Fisher kernel becomes:

$$\mathcal{K}^{\text{PLSI}}\langle d, d' \rangle = \sum_{z \in \mathcal{Z}} \frac{p(z, d)p(z, d')}{p(z)} + \sum_{w \in \mathcal{W}} \hat{p}(d, w)\hat{p}(d', w) \sum_{z \in \mathcal{Z}} \frac{p(z|d, w)p(z|d', w)}{p(w|z)}. \quad (2)$$

Note that the above assumption differs slightly from the one made in [1]: $\sum_w \frac{\hat{p}(w|d)}{p(w|d)} p(w|z) \approx 1$. As the parameters of PLSI are obtained by maximizing the log-likelihood $l_d^{\text{PLSI}}(\theta) = \sum_{d, w} n(d, w) \log p(d, w)$, they also minimize the KL-divergence between $\hat{p}(d, w)$ and $p(d, w)$. PLSI thus aims at finding estimates $p(d, w)$ that approximate $\hat{p}(d, w)$, which naturally lead us to replace the approximation in [1] by the above one.

The above Fisher kernels suffer from the fact that the word counts, $n(d, w)$ or $\hat{p}(d, w)$, are not normalized by the document length, so that documents are compared on the basis of raw counts instead of frequencies, as results from the IR community suggest (e.g. [10]). In order to get to normalized versions, researchers have considered different pseudo-likelihood functions, either a normalized expected log-likelihood for Naive Bayes [9], or a likelihood normalized by the document length for PLSI [1]. There is however no real theoretical justification for deriving Fisher kernels from the former two pseudo-likelihood functions.

3 Underlying Vector Spaces

Both (1) and (2) exhibit two distinct contributions: one from the class probabilities $p(z)$, and the other from the conditional word probabilities $p(w|z)$. The associated kernels implement a dot product in a feature space, the dimensions of which correspond to the former sets of parameters. There are fundamentally two different types of vector spaces involved in this computation: First, a *class-related* K -dimensional subspace, the dimensions of which correspond to the class-probabilities, then K M -dimensional *word-related* subspaces, the dimensions of which correspond to the vocabulary words.

In the class-related vector space, documents d , with components $[p(z_k|d)]$ or $[p(z_k, d)]$, $k \leq K$, are compared on the basis of the latent topics they contain.

In each of the K word-related subspaces, vectors of word counts $[n(d, w_j)]$ or $[\hat{p}(d, w_j)]$, $j \leq M$, weighted with the word and document contributions to the current class z_k ($p(w_j|z_k)$ and $p(z_k|d)$ or $p(z_k|d, w_j)$), are compared. The representations of a document d in these K different subspaces only differ on the weighting of its standard bag-of-words representation, $(n(d, w_1), \dots, n(d, w_M))$, here based on raw counts. Note that neither this representation nor the weighting used in each subspace makes use of any document length normalization, so that longer documents (i.e. containing more words) tend to yield higher similarities with other documents. To compensate this effect, the IR community rather relies on word frequency vectors, leading to the basic document representation $(\hat{p}(w_1|d), \dots, \hat{p}(w_M|d))$, which can then be reweighted in different ways (e.g. with an IDF coefficient, the role of which is here played by the factors $p(w_j|z_k)$) [10].

The above considerations lead us to revisit the original kernels (1) and (2), and introduce a normalization by the document length only in the K word-related vector spaces. This leads to:

$$\mathcal{K}^{\text{NB}'} \langle d, d' \rangle = \sum_{z \in \mathcal{Z}} \frac{p(z|d)p(z|d')}{p(z)} + \sum_{w \in \mathcal{W}} \hat{p}(w|d)\hat{p}(w|d') \sum_{z \in \mathcal{Z}} \frac{p(z|d)p(z|d')}{p(w|z)}, \quad (3)$$

$$\mathcal{K}^{\text{PLSI}'} \langle d, d' \rangle = \sum_{z \in \mathcal{Z}} \frac{p(z, d)p(z, d')}{p(z)} + \sum_{w \in \mathcal{W}} \hat{p}(w|d)\hat{p}(w|d') \sum_{z \in \mathcal{Z}} \frac{p(z|w, d)p(z|w, d')}{p(w|z)}. \quad (4)$$

Interestingly, the form we arrive at for the Naive Bayes model is equivalent to the one derived in [9], but originates from the actual Fisher kernel. For PLSI, the form we obtain resembles the one obtained in [1], but involves a joint, rather than a conditional, (document, topic) distribution in the *class-related* vector space (first term).

4 Impact of the Fisher Information matrix

As already noticed, most authors approximate the Fisher information matrix $G(\theta)$ by the identity matrix. However, theoretical and experimental considerations show that this approximation is not entirely founded for the models here considered, all the more so that the final form we consider integrates normalization by the document length. The complete computation of the Fisher information matrix however remains too complex to be achieved efficiently in practice. We thus resort to an intermediate form, and assume that the Fisher information matrix could be approximated by a diagonal matrix, which corresponds to the diagonal of the actual Fisher information matrix.

In the case of i.i.d. variables, the Fisher information matrix can be approximated, at a maximum likelihood point, by the following quantity (e.g. [11]): $G(\theta) \approx \sum_{d \in \mathcal{D}} (\nabla_{\theta} l_d(\theta))^T (\nabla_{\theta} l_d(\theta))$. The diagonal of this matrix can be efficiently computed as it amounts, for each coefficient, to sum the square of the Fisher scores of the individual documents: $G(\theta)_{(ii)} \approx \sum_{d \in \mathcal{D}} (\nabla_{\theta} l_d(\theta))_{(ii)}^2$. Using this approximation in the above-defined kernels lead for PLSI to:

$$\begin{aligned} \mathcal{K}^{\text{PLSI}''} \langle d_i, d_n \rangle = & \sum_z \frac{p(z, d_i)p(z, d_n)}{p(z)} \underbrace{\frac{1}{\sum_d \left(\frac{p(z, d)}{\sqrt{p(z)}} \right)^2}}_{\text{contribution from } G(\theta)^{-1}} \\ & + \sum_w \hat{p}(w|d_i)\hat{p}(w|d_n) \sum_z \frac{p(z|w, d_i)p(z|w, d_n)}{p(w|z)} \underbrace{\frac{1}{\sum_d \left(\hat{p}(w, d) \frac{p(z|w, d)}{\sqrt{p(w|z)}} \right)^2}}_{\text{contribution from } G(\theta)^{-1}} \end{aligned} \quad (5)$$

and to a similar formula for the Naive Bayes model, where the first term is divided by $\sum_d \left(\frac{p(z|d)}{\sqrt{p(z)}} \right)^2$ instead (conditionnal instead of joint).

5 Experimental Results

5.1 Single Kernel

As done in [1], we have tested our approach on four standard IR benchmark collections [12]: CISI (1460 doc.), CRAN (1400 doc.), MED (1033 doc.) and CACM (3204 doc.). For all the tests, documents have been lemmatized and stemmed. As a baseline, we use a standard *tf-idf* kernel, which roughly corresponds to the kernels we have defined (eq. (3) and (4)) for $K = 1$.

In order to validate the different developments presented in the former sections, we have tested the different forms of the Naive Bayes and PLSI kernels for different values of K . The obtained results are shown in Table 1.³

The first conclusion we can draw from these results is that the normalization based on the underlying vector spaces (VS) significantly improves the performance of the kernel, and leads to results on par with (CACM & MED) and significantly better than (CISI & CRAN) the standard *tf-idf* approach. The second conclusion is that taking into account the diagonal Fisher information matrix, as presented in Sect. 4, further improves the Naive Bayes kernel. However, for PLSI, the use of the diagonal Fisher information matrix does not yield further improvements⁴. This difference is possibly due to the fact that Naive Bayes yields “harder” assignments of documents to classes (one class only for each document) and is thus expected to have less correlations between its parameters. The diagonal of the Fisher information matrix should thus capture most of the information in this case, which does not seem to be true for PLSI.

5.2 Combination of Kernels

One of the major issues with the use of kernels derived from latent class models is the determination of the number of classes. Two main strategies can address this problem: either resort to model selection techniques to find the optimal value of K , or combine kernels obtained with different values of K . The results presented below follow the latter approach. They are based on a simple linear combination $\mathcal{K}_{\text{full}}\langle d, d' \rangle = \sum_r \alpha_r \mathcal{K}_r\langle d, d' \rangle$, where r refers to different values for K .

The weights α_r are determined so as to maximize the R-Precision of the complete kernel $\mathcal{K}_{\text{full}}\langle d, d' \rangle$. In order to keep the computation reasonable, we considered only three kernels: $\mathcal{K}_1$, $\mathcal{K}_{16}$ and $\mathcal{K}_{64}$. The value $K = 1$ was chosen since it yields a kernel similar to a simple *tf-idf* kernel, the good behavior of which is exemplified in Table 1, and was also retained in previously reported experiments [5]. As the combination we rely on should not decrease the results of any of the individual kernels (the weights for the others could be set to 0), we expected

³ Notice that we used the more usual R-Precision measure instead of the quite unusual 9-pt average precision used in [5] and [1]. Although correlated, the results vary in their absolute values.

⁴ A Wilcoxon test shows in this case that the lines VS and DFIM are not significantly different, except for CISI with $K = 32$, for which VS is superior to DFIM.

Table 1. Comparison of the R-Precision for different kernels for both Naive Bayes and PLSI: VS refers to the normalization associated with the different underlying vector spaces, DFIM refers to the use of the diagonal Fisher information matrix. Numbers in bold correspond to the best results, significantly better than the original kernels, as measured with a Wilcoxon test at 1%.

Kernel		CISI		CRAN		MED		CACM	
		$\mathcal{K}_{32}$	$\mathcal{K}_{64}$	$\mathcal{K}_{32}$	$\mathcal{K}_{64}$	$\mathcal{K}_{32}$	$\mathcal{K}_{64}$	$\mathcal{K}_{32}$	$\mathcal{K}_{64}$
tf-idf ($\simeq \mathcal{K}_1$)		.1862		.2381		.4258		.2169	
Naive Bayes									
original	(eq. 1)	.0591	.0575	.0857	.0804	.1131	.1176	.0381	.0388
vs	(eq. 3)	.0834	.0857	.1962	.1940	.1518	.1470	.1202	.1338
DFIM		.1017	.0998	.2238	.2005	.1554	.1570	.1411	.1548
PLSI									
Hofmann's (eq. 6 & 7 in [1])		.1013	.1164	.1631	.1505	.3146	.3897	.1100	.1189
vs	(eq. 4)	.1971	.1779	.2961	.2904	.4203	.4283	.2118	.1787
DFIM	(eq. 5)	.1650	.1679	.2918	.2971	.4023	.4285	.1871	.2244

Table 2. R-Precision for the combination $\mathcal{K}_{\text{full}} = \alpha_1 \mathcal{K}_1 + \alpha_{16} \mathcal{K}_{16} + \alpha_{64} \mathcal{K}_{64}$, with the corresponding weight values, for the original Hoffman's and DFIM formulations of the PLSI model

	Hofmann's					DFIM				
	α_1	α_{16}	α_{64}	$\mathcal{K}_{\text{full}}$	$\text{best}(\mathcal{K}_1, \mathcal{K}_{16}, \mathcal{K}_{64})$	α_1	α_{16}	α_{64}	$\mathcal{K}_{\text{full}}$	$\text{best}(\mathcal{K}_1, \mathcal{K}_{16}, \mathcal{K}_{64})$
CISI	0.9	0.1	0.0	.2246	.1862	0.1	0.5	0.4	.2003	.1862
CRAN	0.8	0.2	0.0	.3110	.2381	0.2	0.4	0.4	.3395	.2971
MED	0.9	0.1	0.0	.4754	.4248	0.1	0.5	0.4	.4790	.4285
CACM	0.9	0.1	0.0	.2133	.2169	0.0	0.6	0.4	.2350	.2244

that considering additional kernels would improve the results of the *tf-idf* kernel. To assess this, we arbitrarily chose the values $K = 16$ and $K = 64$ for the additional two kernels. We then first normalized the kernels, to have similarities in $[0...1]$ and, setting the constraint $\alpha_1 + \alpha_{16} + \alpha_{64} = 1$, we computed all possible values of the kernel combination by varying α_1 and α_{16} in $[0...1]$ (see Fig. 1). We tested this approach on PLSI, as this model yields the best individual kernel performance. Table 2 summarizes the results obtained.

The first conclusion that can be drawn from these results is that the combination, especially when it is coupled with the DFIM version, has a positive effect on the results, as the R-Precision has improved for almost all collections. The only collection on which it is not the case is CISI, which is not so surprising as CISI is known to be a “difficult case” for content-only approaches, with only a few common words between queries and documents (leading to usually low precision) [12, pp 91–94].

The second conclusion concerns the role of the kernel with $K = 1$, which is predominant in the original Hofmann's version, but somewhat minor in our DFIM

Fig. 1. Graphical representation of the performance of a combination of three kernels with different number of classes, $K = 1, 16$ and 64 , for different values of the combinations. The horizontal axis corresponds to α_1 (between 0 and 1) and the vertical axis to α_{16} (also between 0 and 1; $\alpha_{64} = 1 - \alpha_1 - \alpha_{16}$). The greyscale represents the different values of the R-Precision, with white for low values and black for the high values.

version. In order to visualize the impact of each kernel on the combination, we plotted the R-Precision on a 2-dimensional space corresponding to the different values for α_1 (x-Axis, $0 \leq \alpha_1 \leq 1$) and α_{16} (y-Axis, $0 \leq \alpha_{16} \leq 1$). The results obtained are displayed in Fig. 1.

Interestingly, it can be seen from these results that our approach is more robust and homogeneous than the original (Hofmann’s) formulation. Indeed, most of the performance of the original formulation is obtained on the $\alpha_{64} = 0$ line (top-left to bottom-right line) and $\alpha_1 \simeq 1$ area (bottom-left part), which shows the predominance of the baseline $K = 1$ system. The peak performance of the DFIM model is, however, obtained in the “average area” ($\alpha_1 \simeq \alpha_{16} \simeq \alpha_{64} \simeq \frac{1}{3}$). Moreover, the area on which the performance is high (dark area) is significantly larger with the DFIM model, which shows that this model is more robust to the values α_r of the combination.

Note however that we did not optimize the combination weights on an independent validation set, but on the entire collection, as, for two of the four collections (MED, 30 queries, and CACM, 52 queries), we did not have enough queries to constitute distinct sets of reasonable size. The robustness of the DFIM model however suggests that the obtained results should be largely independent of the training/validation/test split retained.

6 Conclusion

Starting from the original theory of Fisher kernels and including specific considerations on the underlying vector space for models of documents, we derived new kernel versions for both the Naive Bayes and PLSI models, which we validated experimentally in IR tasks. In these experiments, the new versions outperform the versions previously proposed. The investigation we made on the underlying vector spaces in which kernels are computed, and the different normalizations we introduced in these spaces, yield a general method to adapt Fisher kernels

to different situations and tasks. These considerations also allowed us to justify the form of the Fisher kernel for Naive Bayes used in previous works. We also proposed a new approximation to the inverse Fisher information matrix, based on the diagonal of the empirical Fisher information matrix. Although this new approximation does not improve the Fisher kernel for PLSI, it significantly improves the one for Naive Bayes. In the future, we plan to investigate whether the same approach can be applied to other tasks (as document clustering or categorization) and other “generative” kernels.

References

1. Hofmann, T.: Learning the similarity of documents: An information-geometric approach to document retrieval and categorization. In: *Advances in Neural Information Processing Systems (NIPS)*. Volume 12. (2000) 914–920
2. Jaakkola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In: *Advances in Neural Information Processing Systems (NIPS)*. Volume 11. (1999) 487–493
3. Lewis, D.: Naive (Bayes) at forty: The independence assumption in information retrieval. In: *Proc. of 10th European Conference on Machine Learning (ECML'98)*. Volume 1398 of *Lecture Notes In Computer Science.*, Springer (1998) 4–15
4. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using EM. *Machine Learning* **39** (2000) 103–134
5. Hofmann, T.: Probabilistic latent semantic indexing. In: *Proc. of 22th International Conference on Research and Development in Information Retrieval*. (1999) 50–57
6. Nyffenegger, M.: Similarités textuelles à base de noyaux de Fisher. Master's thesis, Ecole Polytechnique Fédérale de Lausanne, Switerland (2005)
7. Jin, X., Zhou, Y., Mobasher, B.: Web usage mining based on probabilistic latent semantic analysis. In: *Proc. of the 10th ACM SIGKDD International Conference on Knowledge discovery and Data Mining (KDD'04)*. (2004) 197–205
8. Ahrendt, P., Goutte, C., Larsen, J.: Co-occurrence models in music genre classification. In: *IEEE Int. Workshop on Machine Learning for Signal Processing*. (2005)
9. Vinokourov, A., Girolami, M.: A probabilistic framework for the hierarchic organisation and classification of document collections. *Journal of Intelligent Information Systems* **18** (2002) 153–172
10. Salton, G., McGill, M.: *Introduction to Modern Information Retrieval*. McGraw-Hill, New York (1983)
11. McLachlan, G., Peel, D.: *Finite Mixture Models*. Wiley (2000)
12. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: *Modern Information Retrieval*. Addison-Wesley (1999)