Modeling Term Dependencies with Quantum Language Models for IR

Alessandro Sordoni
sordonia@iro.umontreal.ca

Jian-Yun Nie
nie@iro.umontreal.ca

Yoshua Bengio
bengioy@iro.umontreal.ca

DIRO, Université de Montréal
Montréal, H3C 3J7, Québec

ABSTRACT

Traditional information retrieval (IR) models use bag-of-words as the basic representation and assume that some form of independence holds between terms. Representing term dependencies and defining a scoring function capable of integrating such additional evidence is theoretically and practically challenging. Recently, Quantum Theory (QT) has been proposed as a possible, more general framework for IR. However, only a limited number of investigations have been made and the potential of QT has not been fully explored and tested. We develop a new, generalized Language Modeling approach for IR by adopting the probabilistic framework of QT. In particular, quantum probability could account for both single and compound terms at once without having to extend the term space artificially as in previous studies. This naturally allows us to avoid the weight-normalization problem, which arises in the current practice by mixing scores from matching compound terms and from matching single terms. Our model is the first practical application of quantum probability to show significant improvements over a robust bag-of-words baseline and achieves better performance on a stronger non bag-of-words baseline.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords

Density Matrices; Language Modeling; Retrieval Models

1. INTRODUCTION

The quest for the effective modeling of term dependencies has been of central interest in the information retrieval (IR) community since the inception of first retrieval models. However, the gradual shift towards non bag-of-words models is strewn with modeling difficulties. One of the central problems is to find an effective way of representing and scoring documents based on such dependencies. As pointed out by Gao et al. [9], dependencies can be handled in two ways.

The first approach is to extend the dimensionality of the representation space. In early geometrical retrieval models such as the Vector Space Model (VSM), dependencies arising from phrases (compound terms) are represented by defining additional dimensions in the space, i.e. both the phrase and its component single terms are regarded as representation features [8, 21, 28]. For example, *computer architecture* is considered as disjoint from *computer* and *architecture*, which is a strong modeling assumption, and does not take advantage of the semantic relation that generally exists between a compound phrase and its component terms.

The second approach is more principled in such that simple terms are kept as representational units and term dependencies are modeled statistically as joint probabilities, i.e. $p(computer, architecture)$. Proposed dependence models such as n-gram Language Model (LM) for IR [30], biterm LM [31] or the dependence LM [9] adopt such a representation. However, the gain from integrating dependencies was smaller than hoped [35] and it came with higher computational costs due to dependency parsing or n-gram models [13, 30], or unsupervised iterative methods for estimating the joint probability [9].

Recently, non bag-of-words models such Markov random field (MRF) [19], quasi-synchronous dependence model [24] and the query hypergraph model [2] have been proposed. Most of these retrieval models take a log-linear form, which offers a very flexible way of taking into account term dependencies by integrating different sources of evidence, such as proximity heuristics and exact matching. However, the LM is used as a black box to estimate single-term and compound-term influences separately and then the model combines them to compute the final score. We believe that, from a representational point of view, these models have implicitly made a turn back to the first VSM approach in the sense that the dependencies are assumed to represent additional concepts, i.e. atomic units for the purpose of document and query representation, thus disjoint from the component terms [2, 3]. This choice indeed allows for flexible scoring functions. However, the retrieval model boils down to a combination of scores obtained separately from matching single terms and from matching compound dependencies. This is the main cause of the weight-normalization problem [9, 11] which is that a dependency may be counted twice, as a compound and as component terms. In the context of phrases, Sparck Jones et al. note that "the weight of the

SIGIR'13, July 28–August 1, 2013, Dublin, Ireland.

phrase should reflect not the increased odds of relevance implied by its presence as compared to its absence, as a whole unit, but the increased odds compared to the presence of its components words" [11]. When integrating the evidence, the weights for the combination are usually estimated by optimizing a retrieval measure such as mean average precision (MAP). In this sense, a principled probabilistic interpretation of these models is difficult.

The pioneering work by Van Rijsbergen [33] officially formalized the idea that Quantum Theory (QT) could be seen as a "formal language that can be used to describe the objects and processes in information retrieval". The idea of QT as a framework for manipulating vector spaces and probability is appealing. However, the methods that stem from this initial intuition provided only limited evidence about the usefulness and effectiveness of the framework for IR tasks. For example, Piwowarski et al. [25] test if acceptable performance for ad-hoc tasks could be achieved with a quantum approach to IR. The authors represent documents as subspaces and queries as density operators. However, both documents and queries representations are estimated through passage-retrieval like heuristics, i.e. a document is divided into passages and is associated to a subspace spanned by the vectors corresponding to document passages [25]. Different representations for the query density matrix are tested but none of them led to good retrieval performance. Successively, a number of works took inspiration from quantum phenomena in order to relax some common assumption in IR [37, 38]. Zuccon and Azzopardi [38] introduce interference effects into the Probability Ranking Principle (PRP) in order to rank interdependent documents. Although this method achieves good results, it does not make principled use of the quantum probability space and cannot be considered as evidence towards the usefulness of the enlarged probabilistic space. In general, these methods made heuristic use of the concepts of the theory and no clear probabilistic interpretation can be given.

The intrinsic heuristic flavor in preceding approaches motivated some authors to provide evidence to the hypothesis that there exists an IR situation in which classical probabilistic IR fails, or it is severely limited, and it is thus necessary to switch to a more general probabilistic theory [16, 17, 34]. Although these works are theoretically grounded and heavily influenced our general vision of the theory, no clue is given on how to operationalize such results in real-world applications.

In this paper, we propose a novel retrieval framework for modeling term dependencies based on the probabilistic calculus offered by QT. In our model, both single terms and compound dependencies are mathematically modeled as projectors in a vector space, i.e. elementary events in an enlarged probabilistic space. In particular, a compound dependency is represented as a *superposition* event which is a special kind of projector that is neither disjoint from its component terms, nor a joint event. Documents and queries are represented as a sequence of projectors associated to a *quantum language model* (QLM), encapsulated in a particular matrix. The scoring function is a divergence between query and document QLMs. We will show that our model is a generalization of classical unigram LMs. To our knowledge, this work can be seen as the first work to use the quantum probabilistic calculus in order to achieve improvements over state-of-the-art models.

Our contributions are as follows:

1. We propose a novel application of quantum probability to IR.
2. Using this approach, we show significant improvements over a strong baseline bag-of-words model and a strong non bag-of-words model.
3. We propose a new way of representing dependencies without artificially extending the term space and without estimating expensive n-gram probabilities.
4. We show how the new representation of the dependency permits to specify how the dependency behaves with respect to its component terms.
5. In our model, the dependency information is not integrated in the scoring phase, but in the estimation phase. Hence, our model does not suffer the weight-normalization problem.

2. A BROADER VIEW ON PROBABILITY

2.1 The Quantum Sample Space

In quantum probability, the probabilistic space is naturally encapsulated in a vector space, specifically a Hilbert space, noted $\mathbb{H}^n$, but for the sake of simplicity, in this paper we limit ourselves to finite real spaces, noted $\mathbb{R}^n$. We will be using Dirac's notation restricted to the real field, for which a unit vector $\vec{u} \in \mathbb{R}^n, \|\vec{u}\|_2 = 1$ and its transpose $\vec{u}^\top$ are respectively written as a *ket* $|u\rangle$ and a *bra* $\langle u|$. Using this notation, the projector onto the direction u writes as $|u\rangle\langle u|$. The inner product between two vectors writes as $\langle u|v\rangle$. Moreover, we note by $|e_i\rangle$ the elements of the standard basis in $\mathbb{R}^n$, i.e. $|e_i\rangle = (\delta_{1i}, \ldots, \delta_{ni})^\top$, where $\delta_{ij} = 1$ iff $i = j$.

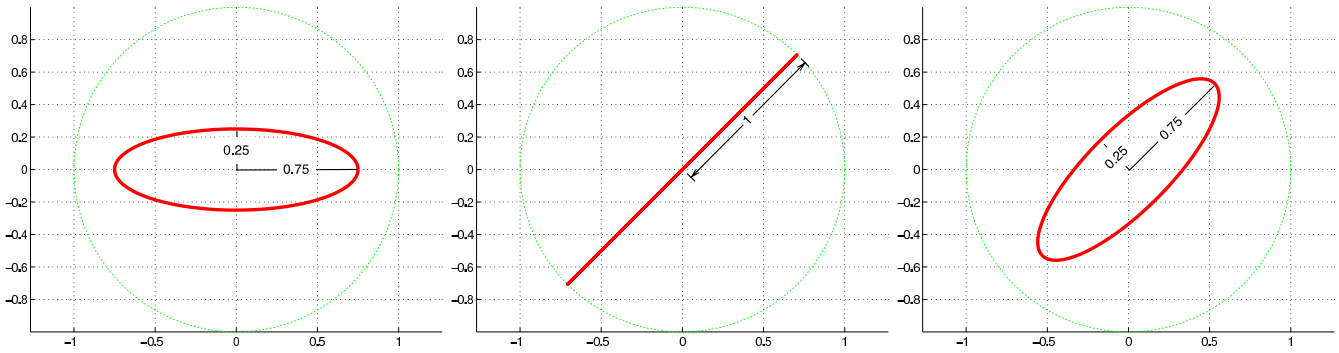
Events are no more defined as subsets but as subspaces, more specifically as projectors onto subspaces [23, 34]. Given a 1-dimensional subspace spanned by a ket $|u\rangle$, the projector onto the unit norm vector $|u\rangle$, $|u\rangle\langle u|$, is an elementary event of the quantum probability space, also called a *dyad*. A dyad is always a projector onto a 1-dimensional space. Given the bijection between subspaces and projectors, it is correct to state that $|u\rangle$ is itself an elementary event. For example, if $n = 2$, the quantum elementary events $|e_1\rangle = (1, 0)^\top$, $|f_1\rangle = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^\top$, can be represented by the following dyads:

$$|e_1\rangle\langle e_1| = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, |f_1\rangle\langle f_1| = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}. \quad (1)$$

Generally, any ket $|v\rangle = \sum_i v_i |u_i\rangle$ is called a *superposition* of the $\{|u_i\rangle\}$ where $\{|u_1\rangle, \ldots, |u_n\rangle\}$ form an orthonormal basis. In order to see the generalization that is taking place, one has to consider that in $\mathbb{R}^n$ there is an infinite number of vectors even if the dimension n is finite. Hence, contrary to the classical case, an infinite number of elementary events can be defined.

2.2 Density Matrices

A quantum probability measure μ is the generalization of a classical probability measure such that (i) for every dyad $|u\rangle\langle u|$, $\mu(|u\rangle\langle u|) \in [0, 1]$ and (ii) it reduces to a classical probability measure for any orthonormal basis $\{|u_1\rangle, \ldots, |u_n\rangle\}$, i.e. $\sum_i \mu(|u_i\rangle\langle u_i|) = 1$. Gleason's Theorem [10] states that,



(a) $\rho = \left(\begin{smallmatrix} 0.75 & 0 \\ 0 & 0.25 \end{smallmatrix}\right)$ (b) $\rho = \left(\begin{smallmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{smallmatrix}\right)$ (c) $\rho = \left(\begin{smallmatrix} 0.5 & 0.25 \\ 0.25 & 0.5 \end{smallmatrix}\right)$

Figure 1: The ellipses depict the set of points $\{\rho|u\rangle : |u\rangle \in \mathbb{R}^2\}$. The eigenvalues of ρ define how much each ellipse is stretched along the corresponding eigenvectors. To the left, ρ corresponds to a classical probability distribution. To the center, ρ is a pure state, thus the ellipse degenerates along the eigenvector corresponding to its unit eigenvalue. To the right, a general density matrix for which we vary both the eigenvalues and the eigensystem.

for any real vector space with dimension greater than 2, there is a one-to-one correspondence between quantum probability measures μ and *density matrices* ρ. The form of this correspondence is given by:

$$\mu_\rho(|v\rangle\langle v|) = \mathrm{tr}(\rho|v\rangle\langle v|). \tag{2}$$

A real density matrix is symmetric, $\rho = \rho^\top$, positive semidefinite, $\rho \geq 0$, and of trace 1, $\mathrm{tr}\,\rho = 1$[1]. From now on, the set of $n \times n$ real density matrices would be noted $\mathcal{S}^n$.

By Gleason's theorem, a density matrix can be seen as the proper quantum generalization of a classical probability distribution. It assigns a quantum probability to each one of the infinite dyads. For example, the density matrix:

$$\rho = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}, \tag{3}$$

assigns probabilities $\mathrm{tr}(\rho|e_1\rangle\langle e_1|) = 0.5$ and $\mathrm{tr}(\rho|f_1\rangle\langle f_1|) = 1$. Hence, the event $|f_1\rangle\langle f_1|$ is certain and still there is non-classical uncertainty on $|e_1\rangle\langle e_1|$. Only if $\{|u_1\rangle, \ldots, |u_n\rangle\}$ form an orthonormal system of $\mathbb{R}^n$ can the dyads $|u_i\rangle\langle u_i|$ be understood as disjoints events of a classical sample space, i.e. their probabilities sum to one. The relation that ties $|e_1\rangle\langle e_1|$ and $|f_1\rangle\langle f_1|$ is purely geometrical and cannot be expressed using set theoretic operations.

Any classical discrete probability distribution can be seen as a mixture over n elementary points, i.e. a parameter $\vec{\theta} = (\theta_1, \ldots, \theta_n)$, where $\theta_i \geq 0$ and $\sum_i \theta_i = 1$. The density matrix is the straightforward generalization of this idea by considering a mixture over orthogonal dyads $\rho = \sum_i \upsilon_i |u_i\rangle\langle u_i|$ where $\upsilon_i \geq 0$ and $\sum_i \upsilon_i = 1$. Given a density matrix ρ, one can find the components dyads by taking its eigendecomposition and building a dyad for each eigenvector. We note such decomposition by $\rho = R\Lambda R^\top = \sum_{i=1}^n \lambda_i |r_i\rangle\langle r_i|$, where $|r_i\rangle$ are the eigenvectors and λ_i their corresponding eigenvalues. This decomposition always exists for density matrices [23].

Conventional probability distributions can be represented by diagonal density matrices. The sample space corresponds to the standard basis $\mathcal{E} = \{|e_i\rangle\langle e_i|\}_{i=1}^n$. Hence, the density matrix corresponding to the parameter $\vec{\theta}$ above can be represented as a mixture over $\mathcal{E}$, i.e. $\rho_\theta = \mathrm{diag}(\vec{\theta}) = \sum_i \theta_i |e_i\rangle\langle e_i|$.

[1]The trace is equal to the sum of the diagonal terms in a matrix.

Consider a vocabulary of two terms $\mathcal{V} = \{a, b\}$. A unigram language model $\vec{\theta} = (0.75, 0.25)$ defined on $\mathcal{V}$ is represented by:

$$\rho_\theta = \frac{3}{4}|e_a\rangle\langle e_a| + \frac{1}{4}|e_b\rangle\langle e_b| = \begin{pmatrix} 0.75 & 0 \\ 0 & 0.25 \end{pmatrix}.$$

Hence, term projectors are orthogonal, i.e. terms correspond to disjoint events. For example, the probability of the term a is computed by $\mathrm{tr}(\rho_\theta|e_a\rangle\langle e_a|) = 0.75$. As conventional probability distributions are restricted to the identity eigensystem, they differ in their eigenvalues, which correspond to diagonal entries. On the contrary, general density matrices can differ also in the eigensystem. For example, the density matrix ρ of Eq. 3 has eigenvector $|f_1\rangle = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^\top$ with eigenvalue 1 and the eigenvector $|f_2\rangle = (\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}})^\top$ with eigenvalue 0. Hence, it can be represented as a one-element mixture containing the projector $\rho = |f_1\rangle\langle f_1|$. When the mixture weights are concentrated into a single projector, the corresponding density matrix is called *pure state*. Otherwise, it is called *mixed state*.

When defined over $\mathbb{R}^n$, density matrices can be seen as ellipsoids, i.e. deformations of the unit sphere (Figure 1) [34]. Classical probability distributions, i.e. diagonal density matrices, are ellipsoids stretched along the identity eigensystem. As quantum probability has access to an infinite number of eigensystems, the ellipsoid can be "rotated", i.e. defined on a different eigensystem. In this work, we will use this additional feature in order to build a more reliable representation of documents and queries taking into account more complex information than single terms.

3. QUANTUM LANGUAGE MODELS

The approach Quantum Language Modeling (QLM) retains the classical Language Modeling for IR as a special case. Hereafter, we will present in details the quantum counterpart of unigram language models. Although it is not explicitly developed in this paper, we argue that arbitrary n-gram models could be modeled as well.

3.1 Representation

In classical bag-of-words language models, a document d is represented by a sequence of i.i.d. term events, i.e.

$\mathcal{W}_d = \{w_i : i = 1, \ldots, N\}$, where N is the document length. Each w_i belongs to a sample space $\mathcal{V}$, corresponding to the vocabulary, of size n. It is assumed that such sequences correspond to a sample from an unknown distribution $\vec{\theta}$ over the vocabulary $\mathcal{V}$, for which we want to gain insight.

A quantum language model assigns quantum probabilities to arbitrary subsets of the vocabulary. It is parametrized by an $n \times n$ density matrix ρ, $\rho \in \mathcal{S}^n$, where n is the size of the vocabulary $\mathcal{V}$. In QLM, a document d is considered as a sequence of M quantum events associated with a density matrix ρ:

$$\mathcal{P}_d = \{\Pi_i : i = 1, \ldots, M\}, \tag{4}$$

where each Π_i is a general dyad $|u\rangle\langle u|$ and represents a subset of the vocabulary. Note that the number of dyads M can be different from N, the total number of terms in the document. The sequence $\mathcal{P}_d$ is constructed from the observed terms $\mathcal{W}_d$: we have to define how to map subsets of terms to projectors. Separating the observed text from the observed projectors constitutes the main flexibility of our model. In what follows, we define a way of mapping single terms and arbitrary dependencies to quantum elementary events. Formally, we seek to define a mapping $m : \mathcal{P}(\mathcal{V}) \to \mathcal{L}(\mathbb{R}^n)$, where $\mathcal{P}(\mathcal{V})$ is the powerset of the vocabulary and $\mathcal{L}(\mathbb{R}^n)$ is the set of dyads on $\mathbb{R}^n$. As an initial assumption, we set $m(\emptyset) = \mathbb{O}$, where $\mathbb{O}$ is the projector onto the zero vector.

3.1.1 Representing Single Terms

In Section 2.2, we showed that unigram sample spaces can be represented as the set of projectors on the standard basis $\mathcal{E} = \{|e_i\rangle\langle e_i|\}_{i=1}^n$ and unigram language models can be represented as mixtures over $\mathcal{E}$, i.e. diagonal matrices. Therefore, a straightforward mapping from single terms to quantum events is:

$$m(\{w\}) = |e_w\rangle\langle e_w|, \tag{5}$$

where $w \in \mathcal{V}$. This choice associates the occurrence of each term to a dyad $|e_w\rangle\langle e_w|$, and these dyads form an orthonormal basis. Hence, occurrences of single terms are still represented as disjoint events. Consider $n = 3$ and $\mathcal{V} = \{$*computer*, *architecture*, *games*$\}$. If $\mathcal{W}_d = \{$*computer*, *architecture*$\}$ and one applies m to each of the terms, the sequence of corresponding projectors is $\mathcal{P}_d = \{\mathcal{E}_{computer}, \mathcal{E}_{architecture}\}$ where $\mathcal{E}_w = |e_w\rangle\langle e_w|$:

$$\mathcal{E}_{computer} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \mathcal{E}_{architecture} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{6}$$

Note that if we decide to observe only single terms, $\mathcal{P}_d$ turns out to be the quantum counterpart of classical observed terms $\mathcal{W}_d$, i.e. $M = N$.

3.1.2 Representing Dependencies

In this paper, by dependency, we mean a relationship linking two or more terms and we represent such an entity abstractly by a subset of the vocabulary, i.e. $\kappa = \{w_1, \ldots, w_K\}$. We define the following mapping for an arbitrary dependency κ:

$$m(\kappa) = m(\{w_1, \ldots, w_K\}) = |\kappa\rangle\langle\kappa|, \; |\kappa\rangle = \sum_{i=1}^{K} \sigma_i |e_{w_i}\rangle, \tag{7}$$

where the coefficients $\sigma_i \in \mathbb{R}$ must be chosen such that $\sum_i \sigma_i^2 = 1$, in order to ensure the proper normalization of $|\kappa\rangle$. The well-defined dyad $|\kappa\rangle\langle\kappa|$ is a *superposition* event. As we showed in Section 2.2, superposition events are justifiable only in the quantum probabilistic space. They are neither disjoint from their constituents $|e_{w_i}\rangle\langle e_{w_i}|$ nor do they solely constitute joint events in the sense of n-grams: here, the compound dependency is not considered as an additional entity, as done in previous models [2, 3, 19, 21]. The proposed mapping allows for the representation of relationships within a group of terms by creating a new quantum event in the same n-dimensional space.

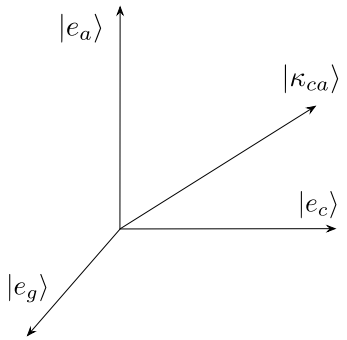


Figure 2: The dependency κ_{ca} is modeled as a projector onto $|\kappa_{ca}\rangle$, i.e. as a superposition event.

In addition, superposition events come with a flexible way in quantifying how much evidence the observation of dependency κ brings to its component terms. This is achieved by changing the distribution of the σ_i: if one wants to attempt a classical interpretation, the σ_i can be viewed as relative pseudo-counts, i.e. observing $|\kappa\rangle\langle\kappa|$ adds fractional occurrence to the events of its component terms $|e_{w_i}\rangle\langle e_{w_i}|$. To our knowledge, until now this feature has been only modeled heuristically, or not modeled at all. In our framework, it fits nicely in the quantum probabilistic space by specifying how a compound dependency event and its constituent single terms events are related.

As an example, one could model the compound dependency between *computer* and *architecture*, $\kappa_{ca} = \{$*computer*, *architecture*$\}$, by the dyad $\mathcal{K}_{ca} = |\kappa_{ca}\rangle\langle\kappa_{ca}|$, where $|\kappa_{ca}\rangle = \sqrt{2/3}|e_c\rangle + \sqrt{1/3}|e_a\rangle$ (Figure 2). With respect to the example taken above, the event is represented by the matrix:

$$\mathcal{K}_{ca} = \begin{pmatrix} \frac{2}{3} & \frac{\sqrt{2}}{3} & 0 \\ \frac{\sqrt{2}}{3} & \frac{1}{3} & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{8}$$

The superposition coefficients entail that observing $\mathcal{K}_{ca}$ adds more evidence to $|e_c\rangle\langle e_c|$ than to $|e_a\rangle\langle e_a|$.

3.1.3 Choosing When and What to Observe

Once we have defined the mapping m, one must ask three questions:

1. Which compound dependencies to consider?
2. When does such a compound dependency hold in a document?
3. When the compound dependency is detected, should we also consider the projectors for its subsets as observed events?

Regarding the first question, one may (a) use a dictionary of phrases or frequent n-grams, or (b) assume that any

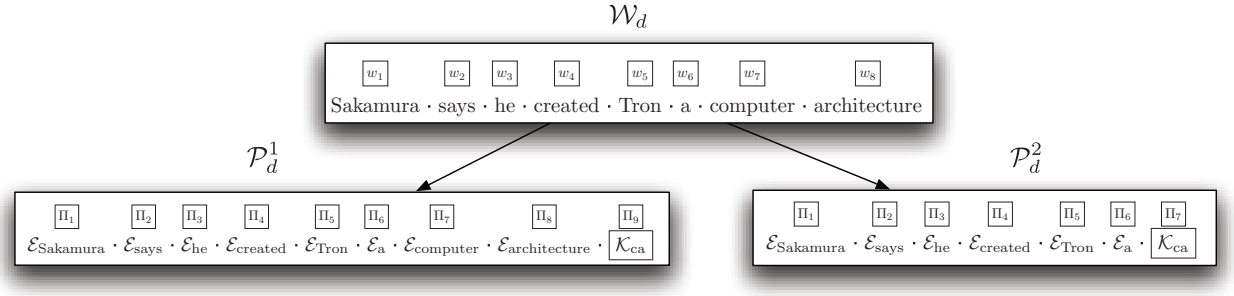


Figure 3: Two possible quantum sequences $\mathcal{P}_d^i$ of an excerpt $\mathcal{W}_d$ from a TREC collection. The observation of *computer architecture* is associated to a superposition projector $\mathcal{K}_{ca} = |\kappa_{ca}\rangle\langle\kappa_{ca}|$ while $\mathcal{E}_w = |e_w\rangle\langle e_w|$ are classical projectors. For $\mathcal{P}_d^2$ we observed only the compound while in $\mathcal{P}_d^1$ we also added its subsets.

subset of terms that appear in short queries are candidate compound dependencies to capture. In this paper, we want to make the approach as independent as possible of any linguistic resource. So the second approach (b) is used. This will also allow us to make a fair comparison with the previous approaches using the same strategy (such as the MRF model [19]).

The second question regards whether such selected compound dependencies hold in a given document. In other words, one has to decide when to add the selected dependency projector into a document sequence $\mathcal{P}_d$. This can be done for example by assuming that the components terms in the dependency appear as a bigram in a document, as biterm or in a unordered window of L terms. Convergent evidence from different works [1, 12, 14, 18, 31, 36] confirms that proximity is a strong indicator of dependence. Therefore, in this work we choose to detect a dependency if its component terms appear in a fixed-window of length L.

The third question regards how to apply the mapping m and can be more easily understood by a practical example. Consider a document $\mathcal{W}_d = \{computer, architecture\}$ and a query $\mathcal{W}_q = \{computer, architecture\}$. Once the dependency $\kappa_{ca} = \{computer, architecture\}$ has been detected in the document, i.e. the component terms appear next to each other, one can further decide:

1. to map only the dependency, i.e. $\mathcal{P}_d = \{\mathcal{K}_{ca}\}$,
2. to map both the dependency and the component terms, i.e. $\mathcal{P}_d = \{\mathcal{E}_{computer}, \mathcal{E}_{architecture}, \mathcal{K}_{ca}\}$.

These two choices are illustrated in Figure 3. The first choice is a highly non-classical one because it completely steals the occurrence of its component terms. Nevertheless, it becomes a valid choice in our framework. Differently from classical approaches, the fact that we only consider a count for the compound *computer architecture* does not mean that we assume that the terms *computer* and *architecture* do not occur. The dependency event is not disjoint from the single term events, and its occurrence partially entails the occurrence of its component terms. However, this choice is more dangerous because it over-penalizes the component terms: we should know very precisely *when* such a strong dependency is observed and *which coefficients* to assign to it.

The second choice is implicitly done in current dependency models and is at the basis of the weight-normalization problem. From this point of view, the sequence $\mathcal{P}_d$ could be seen as composed by concepts as recently formalized by Bendersky et al. [2, 3]. However, there are crucial differences from that work: (1) we give a clear probabilistic status to such concepts and (2) we do not assume that concepts are atomic units of information, completely unrelated from each other. In classical dependence models, single terms and compound dependencies are scored separately and then the scores are combined together [2, 19, 35]. A critical aspect of such models is that the occurrence of the phrase *computer architecture* will be counted twice - as single terms and as a compound. That is why the score on compound dependencies must be reweighed before integrating it with the independence score [9, 11, 19]. Contrary to classical models, our model does not suffer from such a problem because the evidences brought by the compound dependency as a whole and by its component terms are integrated in the estimation phase. Even if not reported explicitly in the experiments section, conducted experiments show that including projectors for both the dependency and its subsets is much more effective for the ad-hoc task evaluated here and thus this strategy will be preferred throughout this paper. In addition, an algorithm building the sequence of projectors from the document sequence will be presented in Section 4.3.1.

3.2 Estimation

3.2.1 Maximum Likelihood Estimation

Given that a document is represented by a set of observed projectors, one has to find ways to learn a quantum language model ρ to associate with a document. In QT, a number of objective functions have been proposed to estimate an unknown density matrix from a set of projectors: Linear Inversion [23] and Hedged ML [4] are notorious examples. In this work, we use the Maximum Likelihood (ML) formulation proposed in [15], because (1) it can easily be seen as a quantum generalization of a classical likelihood function (2) contrary to linear inversion, ML generates a well-defined density matrix, i.e. $\rho \in \mathcal{S}^n$, and (3) proposed estimation methods remain computationally affordable in high-dimensional spaces.

Given the observed projectors $\mathcal{P}_d = \{\Pi_1, \ldots, \Pi_M\}$ for document d, we define as training criterion for the quantum language model ρ the maximization of the following product proposed in [15] and corresponding in the unigram case to a proper likelihood:

$$\mathcal{L}_{\mathcal{P}_d}(\rho) = \prod_{i=1}^{M} \text{tr}(\rho\Pi_i). \tag{9}$$

The estimate $\widehat{\rho}$ can be obtained by approximately solving the following maximization problem:

$$\begin{aligned} \underset{\rho}{\text{maximize}} \quad & \log \mathcal{L}_{\mathcal{P}_d}(\rho) \\ \text{subject to} \quad & \rho \in \mathcal{S}^n. \end{aligned} \tag{10}$$

This maximization is difficult and must be approximated by using iterative methods. In [15], the following iterative scheme is proposed, also called the "$R\rho R$ algorithm". One introduces the operator:

$$R(\rho) = \sum_{i=1}^{M} \frac{1}{\text{tr}(\rho \Pi_i)} \Pi_i, \tag{11}$$

and updates an initial density matrix $\widehat{\rho}_{(0)}$ by applying repetitive iterations:

$$\widehat{\rho}_{(k+1)} = \frac{1}{Z} R(\widehat{\rho}_{(k)}) \widehat{\rho}_{(k)} R(\widehat{\rho}_{(k)}), \tag{12}$$

where, $Z = \text{tr}(R(\widehat{\rho}_{(k)})\widehat{\rho}_{(k)}R(\widehat{\rho}_{(k)}))$ is a normalization factor in order to ensure that $\widehat{\rho}_{(k+1)}$ respects the constraint of unitary trace [15]. Despite the $R\rho R$ algorithm being a quantum generalization of the well-behaving Expectation Maximization (EM) algorithm, the likelihood is not guaranteed to increase at each step because the nonlinear iteration may overshoot, similarly to a gradient descent algorithm with a too big step size. Characterizing such situations still remains an open problem [27]. In this work, in order to ensure convergence, if the likelihood is decreased at $k+1$, we use the following damped update:

$$\widetilde{\rho}_{(k+1)} = (1-\gamma)\widehat{\rho}_{(k)} + \gamma \widehat{\rho}_{(k+1)}, \tag{13}$$

where $\gamma \in [0, 1)$ controls the amount of damping and is optimized by linear search in order to ensure the maximum increase of the training objective[2]. As $\mathcal{S}^n$ is convex [23], $\widetilde{\rho}_{(k+1)}$ is a proper candidate density matrix. The process stops if the change in the likelihood is below a certain threshold or if a maximum number of iterations is attained.

From an IR point of view, the *metric divergence* problem [22] tells us that the maximization of the likelihood does not mean that the evaluation metric under consideration, such as mean average precision, is also maximized. In the experiments section, we address the two following questions from a perspective closer to IR concerns:

1. Which initial matrix $\widehat{\rho}_{(0)}$ to choose?
2. When to stop the update process?

As the estimation of a quantum document model requires an iterative process, one may believe that the complexity will make the process intractable. In Section 4.5, we provide an analysis of the complexity of the proposed computation, which will show that the process is quite tractable.

3.2.2 Smoothing Density Matrices

The ML estimation presented above suffers from a generalization of the usual zero-probability problem of classical ML, i.e. the estimator assigns zero probability to unseen data [35]. This is also called the zero eigenvalue problem [4]. Bayesian smoothing for density matrices has not yet been proposed. This may be because Bayesian inference in the quantum setting has just started to be the subject of intensive research [5, 34]. In this work, we propose to smooth density matrices by linear interpolation [35]. If $\widehat{\rho}_d$ is a document quantum language model obtained by ML, its smoothed version is obtained by interpolation with the ML collection quantum language model $\widehat{\rho}_c$:

$$\rho_d = (1-\alpha_d)\,\widehat{\rho}_d + \alpha_d\,\widehat{\rho}_c, \tag{14}$$

where $\alpha_d \in [0, 1]$ controls the amount of smoothing. As the set of density matrices $\mathcal{S}^n$ is convex, the resulting ρ_d is a proper density matrix. In this work, we assume that $\alpha_d = \frac{\mu}{(\mu+M)}$, which is the well-known form of the parameter for Dirichlet smoothing [35].

3.3 Scoring

The flexibility of the Kullback Liebler (KL) divergence approach in keeping distinct query and document representations makes it attractive for a candidate scoring function in our new framework. The direct generalization of classical KL divergence was introduced by Umegaki in [32] and is called *quantum relative entropy* or *Von-Neumann* (VN) *divergence*. Given two quantum language models ρ_q and ρ_d for the query and a document respectively, our scoring function is the negative query-to-document VN divergence:

$$\begin{aligned} -\Delta_{VN}(\rho_q \| \rho_d) \quad &= \quad -\text{tr}(\rho_q(\log \rho_q - \log \rho_d)) \\ &\overset{rank}{=} \quad \text{tr}(\rho_q \log \rho_d), \end{aligned} \tag{15}$$

where log applied to a matrix denotes the matrix logarithm, i.e. the classical logarithm applied to the matrix eigenvalues. Rank equivalence is obtained by noting that $\text{tr}(\rho_q \log \rho_q)$ does not depend on the particular document. Denote by $\rho_q = \sum_i \lambda_{qi} |q_i\rangle\langle q_i|$, $\rho_d = \sum_i \lambda_{di} |d_i\rangle\langle d_i|$ the eigendecompositions of the density matrices ρ_q and ρ_d respectively. By substituting into the above equation, the scoring function rewrites as:

$$-\Delta_{VN}(\rho_q || \rho_d) \overset{rank}{=} \sum_i \lambda_{qi} \sum_j \log \lambda_{dj} \langle q_i | d_j \rangle^2. \tag{16}$$

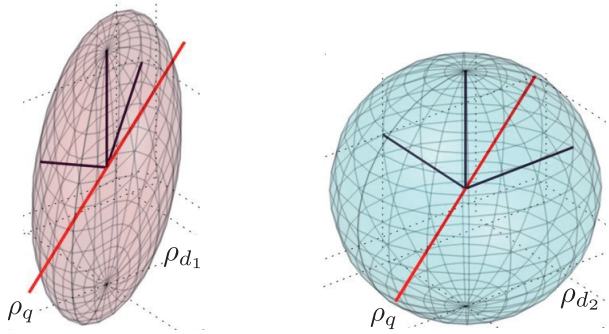
Compared to a classical KL divergence, the additional term $\langle q_i|d_j\rangle^2$ quantifies the difference in the eigenvectors between the two models. Following the representation introduced in Section 2.2, the VN divergence compares two ellipsoids not only by differences in the "shape" but also by differences in the "rotation".

If a VSM-like interpretation is attempted, one can think about $\{|q_i\rangle\}$, $\{|d_j\rangle\}$ as semantic concepts for the query and the document respectively, whereas the vectors of eigenvalues $\vec{\lambda}_q$, $\vec{\lambda}_d$ denote the importance of the corresponding semantic concepts in the two models. The VN divergence offers a way of matching query concepts by analyzing how much such concepts are related to documents concepts, i.e. $\forall i, j$, $\langle q_i|d_j\rangle^2$. Particularly, $\sum_j \langle q_i|d_j\rangle^2 = 1$. Thus, $\langle q_i|d_j\rangle^2$ can be interpreted as the quantum probability associated with the pure state $|q_i\rangle\langle q_i|$ for the elementary event $|d_j\rangle\langle d_j|$, i.e. $\mu_{q_i}(|d_j\rangle\langle d_j|) = \text{tr}(|q_i\rangle\langle q_i|d_j\rangle\langle d_j|) = \langle q_i|d_j\rangle^2$. Hence, one could rewrite Eq. 16 as:

$$-\Delta_{VN}(\rho_q || \rho_d) \overset{rank}{=} \sum_i \lambda_{qi}\, \mathbb{E}_{\mu_{q_i}} \left[\log \vec{\lambda}_d \right]. \tag{17}$$

Therefore, the VN divergence scores a document based on the expectation of how important concept $|q_i\rangle$ is in document d even if it does not appear in it explicitly.

[2]Similar damped updates were successfully used in [26] to improve convergence and stability of the loopy belief propagation algorithm.



(a) $-\Delta_{VN}(\rho_q \| \rho_{d_1}) \propto -.76$ (b) $-\Delta_{VN}(\rho_q \| \rho_{d_2}) \propto -1.06$

o	$\mathcal{W}_o$	$\mathcal{P}_o$
q	{*computer, architecture*}	$\{\mathcal{E}_c, \mathcal{E}_a, \mathcal{K}_{ca}\}$
d_1	{*computer, architecture, and, games*}	$\{\mathcal{E}_c, \mathcal{E}_a, \mathcal{K}_{ca}, \mathcal{E}_g\}$
d_2	{*computer, games, and, architecture*}	$\{\mathcal{E}_c, \mathcal{E}_g, \mathcal{E}_a\}$

Figure 4: A synthetic example of QLM with a vocabulary of $n = 3$ terms. The orthogonal rays are the eigenvectors of the ellipsoids. ρ_q is not smoothed thus degenerates onto a ray. ρ_{d_1} rotates towards the direction of observed query dependencies and is thus ranked higher.

3.4 Final Considerations

The estimation and scoring process of quantum language models retains classical unigram LMs and KL divergence as special cases. The classical unigram LM is recovered by restricting the maximization in Eq. 10 to diagonal density matrices and including into the sequence of projectors $\mathcal{P}_d$ only an orthonormal basis, such as the elements of $\mathcal{E}$. Classical KL divergence is recovered by noting that if ρ_q and ρ_d are diagonal density matrices, they share the same eigensystem. Hence, $|q_i\rangle = |d_i\rangle$ and $\lambda_{qi} = \theta_{qi}$, $\lambda_{di} = \theta_{di}$, where $\vec{\theta}_q$, $\vec{\theta}_d$ are the parameters of classical unigram LMs for the query and the document respectively. In this setting, $\langle q_i | d_j \rangle^2 = 0$ for $i \neq j$ and the VN divergence reduces to classical KL, i.e. $-\Delta_{VN}(\rho_q \| \rho_d) = -\Delta_{KL}(\vec{\theta}_q \| \vec{\theta}_d) \overset{rank}{=} \sum_i \theta_{qi} \log \theta_{di}$.

In Figure 4, we report a synthetic example of the application of the model. We plot the density matrices obtained by the MLE (Section 3.2.1) on the sequence of projectors reported in the table. As usual in ad-hoc tasks, we smooth only the QLMs of the documents. The model corresponding to the query is a projector, i.e. it has two zero eigenvalues, because we did not apply smoothing. If the dependencies are included in the sequence $\mathcal{P}_o$, the MLE rotates the corresponding QLM towards the direction spanned by the observed projector (i.e. $\mathcal{K}_{ca}$). This entails that the model ρ_{d_1} is considered more similar to the query than the model ρ_{d_2} which corresponds to a classical language model.

4. EVALUATION

4.1 Experimental Setup

All the experiments reported in this work were conducted using the open source Indri search engine (version 5.3)[3]. The test collections used are reported in Table 1. We choose the collections in order to vary (1) the collection size and (2) collection type. This will produce a comprehensive test set in order to verify the properties of our approach. All the collections have been stemmed with the Krovetz stemmer. Both documents and queries have been stopped using the standard INQUERY stopword list. For all the methods, the Dirichlet smoothing parameter μ is set to the default Indri value ($\mu = 2500$). The optimization of all the other free parameters for the proposed model and the baselines is done using five-fold cross validation using coordinate ascent [18] with mean average precision (MAP) as the target metric. The performance is measured on the top 1000 ranked documents. In addition to MAP, for newswire collections we report the early precision metric @10 (precision at 10) and for web collections with graded relevance judgements we report the recent ERR@10, which correlates better with click metrics than other editorial metrics [6]. The statistical significance of differences in the performance of tested methods is determined using a two-sided Fisher's randomization test [29] with 25,000 permutations evaluated at $\alpha < 0.05$.

Name	Content	# Docs	Topic Numbers
SJMN	Newswire	90,257	51-150
TREC7-8	Newswire	528,155	351-450
WT10g	Web	1,692,096	451-550
ClueWeb-B	Web	50,220,423	51-200

Table 1: Summary of the TREC collections used to support the experimental evaluation.

4.2 Methodology

Our experimental methodology goes as follows. In a first step, we compare our QLM approach to a unigram Language Modeling baseline (denoted `LM`) based on Dirichlet smoothing [35], which is a strong bag-of-words baseline. This comparison is done by assigning uniform superposition weights to each dependency κ, i.e. $\sigma_i = 1/\sqrt{|\kappa|}$, where $|\kappa|$ is the cardinality of κ (denoted `QLM-UNI`). This step has two main objectives: (1) to test if quantum probability can bring better performance than a standard bag-of-words model and (2) to test if uniform superposition weights are a reasonable baseline setting.

As a second step, we test the proposed model against the strong non bag-of-words MRF model, which has shown to be highly effective especially for large scale web collections [19, 20]. We test the full dependence version of the model (denoted `MRF-FD`) which captures dependencies between all the query terms and thus is the most natural choice for a comparison with our model. However, `MRF-FD` exploits both proximity (#uw) and exact matching (#1). As our model only exploits proximity as an indicator of dependence, we also propose to test the variant `MRF-FD-U`, which is a MRF using only the proximity feature. This could provide interesting insights on how the models score based upon the same evidence.

Finally, we propose a slightly more elaborate version of our model (denoted `QLM-IDF`) in which the superposition weights are no more assumed to be uniform. Instead, we assign to each σ_i the normalized *idf* weight of the corresponding term w_i. The objective is to test if a more reasonable parametrization of superposition weights can improve the retrieval effectiveness.

[3] http://www.lemurproject.org

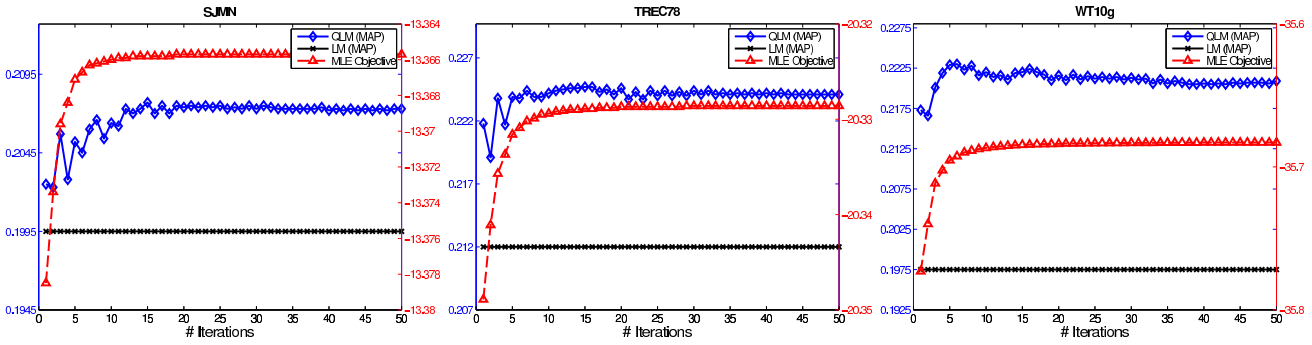


Figure 5: Plots of MAP (QLM-UNI and LM) and MLE objective against the number of updates of the density matrix for SJMN, TREC7-8 and WT10g (left, center and right).

All the results exposed in this paper have been obtained by reranking. We rerank a pool of 20000 documents retrieved using LM in order to make a fair comparison between our method and the baselines.

4.3 Setting up QLM

4.3.1 Building the Sequence of Projectors

Very similarly to MRF-FD, given a query $\mathcal{Q} = \{q_1, \ldots, q_n\}$, we assume that the interesting dependencies to consider correspond to the power set $\mathcal{P}(\mathcal{Q})$[4]. In order to build the set of projectors for the given document we apply Algorithm 1.

Algorithm 1 Builds the sequence $\mathcal{P}_d$ given $\mathcal{W}_d$, $\mathcal{Q}$

Require: $\mathcal{W}_d, \mathcal{Q}$
1: $\mathcal{P}_d \leftarrow \emptyset$
2: **for** $\kappa \in \mathcal{P}(\mathcal{Q})$ **do**
3: **for** $\#(\kappa, \mathcal{W}_d)$ **do**
4: $\mathcal{P}_d \leftarrow \mathcal{P}_d \oplus m(\kappa)$ *%Adds the projector to the sequence*
5: **end for**
6: **end for**
7: **return** $\mathcal{P}_d$

For each dependency κ in $\mathcal{P}(\mathcal{Q})$, the algorithm scans the document sequence $\mathcal{W}_d$. For each occurrence of κ, it adds a projector $m(\kappa)$ to the sequence $\mathcal{P}_d$. The function $\#(\kappa, \mathcal{W}_d)$ returns how many times the dependency κ is observed in $\mathcal{W}_d$. Therefore, the algorithm adds as many projectors as the number of detected compound dependencies. Note that by looping on $\mathcal{P}(\mathcal{Q})$, we are actually implementing the strategy exposed in Section 3.1.3, i.e. adding both the dependence and all of its subsets. Following Section 3.1.3, we choose to parametrize $\#$ as the unordered window operator in Indri ($\#\text{uw}L$). Therefore, a given dependency κ will be detected if the component terms appear in any order in a fixed-window of length $L = l|\kappa|$. This kind of adaptive parametrization of the window length is state-of-the-art for dependence models such as MRF-FD [2, 19]. For all the dependence models, the coordinate ascent for l spans $\{1, 2, 4, 8, 16, 32\}$, which is a robust pool covering different window lengths, including the standard value ($l = 4$) for MRF-FD.

[4]In order to keep the retrieval complexity reasonable both for MRF and QLM, we limit ourselves to query term subsets with at most three terms.

4.3.2 MLE Convergence Analysis

Before doing any comparisons, we answer the questions related to the construction of a quantum language model, i.e. (1) how to initialize $\widehat{\rho}_{(0)}$? (2) when to stop the update process? In order to help the maximum likelihood process to converge faster, we initialize the matrix $\widehat{\rho}_{(0)}$ to the density matrix corresponding to the classical maximum likelihood language model $\vec{\theta}^{ML}$ of the document or query under consideration. This is a diagonal matrix $\widehat{\rho}_{(0)} = \text{diag}(\vec{\theta}^{ML})$. We also tested with the uniform density matrix, as suggested in [15], but we found that the MAP was severely harmed.

In order to address the second question, we analyze the variation of MAP with respect to the maximum number of iterations $n_{it} \in [1, 50]$. The damping factor γ is optimized over the set of values $\Gamma = \{0, 0.1, ..., 0.9\}$. The iterative process stops before n_{it} if the change in the likelihood is below 10^{-4}. In order to check for possible variations due to the collection type, we plot the iteration-MAP curve for two similar collections, i.e. SJMN and TREC7-8, and a web collection, WT10g. We also plot the training objective in Eq. 10 over the set of topics: $\frac{1}{|\mathcal{R}|}\sum_{d\in\mathcal{R}} \log \mathcal{L}_{\mathcal{P}_d}(\widehat{\rho}_d)$, where $\mathcal{R}$ is the multiset of retrieved documents. The trend is shown in Figure 5. Generally, at any number of iterations, the MAP stays significantly above the baseline. It seems that there is a good correlation between likelihood maximization and MAP, although one can note some overfitting at high number of iterations. Capping by $10 \leq n_{it} \leq 20$ seems a good trade-off between likelihood maximization and MAP. However, to provide a fair comparison with the baselines, we choose to include n_{it} as a free parameter to train by coordinate ascent.

4.4 Results

The results discussed in this section are compactly reported in Table 2.

4.4.1 Language Modeling Baseline

From the comparisons with the LM baseline, one can see that QLM-UNI outperforms LM significantly, with relative improvements in MAP going up to 12.1% in the case of WT10g collection and 19.2% for the ClueWeb-B collection. This seems to be in line with the hypothesis formulated in [19], for which dependence models may yield larger improvements for large collections.

The weight-normalization problem seems to be addressed automatically: our model does not need for any combina-

	SJMN		TREC7-8		WT10g		ClueWeb-B	
	P@10	MAP	P@10	MAP	ERR@10	MAP	ERR@10	MAP
LM	.3064	.1995	.4230	.2120	.1068	.1975	.0718	.1003
MRF-FD-U	.3138	.2071	.4350	.2228	.1136	.2097	.0828	.1103
MRF-FD	.3074	.2061	.4460	.2243	.1147	.2146	.0881	.1137
QLM-UNI	**.3181** *(+1.4/+3.5)*	.2077 *(+0.3/+0.8)*	**.4480** *(+3.0/+0.4)*	.2240 *(+0.5/-0.1)*	.1162 *(+2.2/+1.3)*	.2215$^{\alpha\beta}$ *(+5.6/+3.2)*	**.1015**$^{\alpha\beta}$ *(+22.6/+15.2)*	**.1196**$^{\alpha\beta}$ *(+8.4/+5.2)*
QLM-IDF	.3170 *(+1.0/+3.1)*	**.2093** *(+1.1/+1.6)*	.4450 *(+2.3/-0.2)*	**.2254** *(+1.2/+0.5)*	**.1176** *(+3.5/+2.6)*	**.2264**$^{\alpha\beta}$ *(+7.9/+5.5)*	.0997$^{\alpha\beta}$ *(+20.4/+13.1)*	.1189$^{\alpha\beta}$ *(+7.8/+4.5)*

Table 2: Evaluation of the performance for the five methods tested. Best results are highlighted in boldface. Numbers in parentheses indicate relative improvement (%) in MAP over MRF-FD-U/MRF-FD. All the results for dependence models are significant with respect to the baseline LM. The symbols $^{\alpha}$,$^{\beta}$ means statistical significance over MRF-FD-U, MRF-FD respectively.

tion weights. Moreover, it is robust across the folds. From an analysis of the optimal values of the parameters obtained across the different folds, we found that optimal window sizes were $l \in \{1, 2\}$. This can be explained by considering that in the current version of QLM, it is possible to decide if the dependency is detected or not, but the model cannot discriminate its "importance". If one decides to increase l, more inaccurate dependencies will be detected and the performance will be deteriorated. However, even with a larger window size, statistical significance over LM is maintained. From these considerations, we suggest $l = 2$ as a default setting for our model. Finally, the results endorse that our QLM does not need an engineered estimation of superposition weights to perform well.

4.4.2 Markov Random Fields Baseline

As a second test, we report the results obtained for the MRF-FD and MRF-FD-U baselines. These have proved to be very robust non bag-of-words baselines [2, 19, 20]. Contrary to our model, MRF does not handle dependency information in the estimation phase. One has to specify the coefficients $(\lambda_T, \lambda_O, \lambda_U)$ for the combination of dependence and independence scores. To limit per-fold overfitting, for the dependence models, we first train combination parameters $(\lambda_f \in \{0, 0.01, ..., 1\})$ then l for each fold. For MRF-FD-U, we set $\lambda_O = 0$.

Results show that for SJMN and TREC7-8, QLM-UNI, MRF-FD and MRF-FD-U are essentially equivalent. However, for the two Web collections, our model significantly outperforms both MRF variants. On ClueWeb-B, statistical significance is attained for the two reported measures. As conjectured in [19], noisy web collections could be a more discriminative testbed for dependence models. Optimal l values for MRF-FD were very small for SJMN ($l \in \{1, 2\}$) in contrast to the optimal setting for ClueWeb-B ($l \in \{16, 32\}$). In [19], the authors suggest that for homogenous newswire collections a small window is enough to capture useful dependencies, while for large, noisy web collections, a larger span must be set. However, the performances obtained by our model seem to suggest that it can greatly benefit from term dependencies, on a variety of collections, even when a small window size is used. This elucidates the fact that even short range information can be extremely useful if integrated in the estimation phase. In order to get a more comprehensive view on such issues, we trained on the entire set of ClueWeb-B topics three versions of MRF-FD-U, each obtained by clamping a different value of $l \in \{1, 2, 4\}$. The best performing model obtained a MAP of 10.91. It seems that our model can exploit this short range information in a better way than MRF models.

4.4.3 Setting Superposition Weights

Our last test aimed at verifying if a more reasonable setting of the superposition weights could further improve retrieval performance. For a dependency $\{w_1, \ldots, w_K\}$, we set $\sigma_i = \sqrt{idf_{w_i} / \sum_i idf_{w_i}}$. This has the effect of attributing a larger count to the more "important" term in the dependency. QLM-IDF generally increases MAP. However, this is not the case for ClueWeb-B. From a query-by-query analysis, we noticed that QLM-IDF increases the performance for noisy queries by promoting the most "important" terms in unnecessary subsets. For multiword expressions such as ClueWeb-B topics *continental plates* and *rock art*, weighting by *idf* may be misleading by assigning more weight to one of the terms. In this cases, a uniform parametrization is far more effective. This demonstrates that there is still room for improvement by a clever tuning of superposition parameters, for example by leveraging feature functions [2, 3].

4.5 Complexity Analysis

Complexity issues can be tackled by noting that it is not necessary to manipulate $n \times n$ matrices. We associate a dimension for each query term and an additional dimension for a "don't care" term that will store the probability mass for the other terms in the vocabulary. Therefore, a multinomial over n points is reduced to a multinomial over $|\mathcal{Q}| + 1$ points, where $|\mathcal{Q}|$ is the number of unique terms in the query and the additional dimension is simply a relabeling of the other term events. In this way, the QLM to manipulate is $k \times k$, where $k = |\mathcal{Q}| + 1$. The eigendecomposition generally requires $O(k^3)$. The iterative process requires at most $|\mathcal{P}(\mathcal{Q})| = 2^{|\mathcal{Q}|}$ matrix multiplications for the expectation step, where $2^{|\mathcal{Q}|}$ is the maximum number of *unique* projectors in $\mathcal{P}_d$ and 2 matrix multiplications for the maximization step. In the case the likelihood is decreased, $|\Gamma|$ more iterations are done giving a worst-case complexity of $O(n_{it}|\Gamma|2^k + k^3)$, i.e. if each iteration needs damping. We showed that $10 \leq n_{it} \leq 20$ is enough; we use $|\Gamma| = 10$ and k is very small for title queries, which make the process computationally tractable. In practice, we observed that the damping process is very effective and dramatically improves convergence speed. As an example, the mean number of iter-

ations for ClueWeb-B when $n_{it} = 15$ is 7.02 which is orders of magnitude less than $n_{it}|\Gamma| = 150$. Finally, we conjecture that such process could be executed at indexing time, thus eliminating any additional on-line costs.

5. CONCLUSION

In this paper, we presented a principled application of quantum probability for IR. We showed how the flexibility of vector spaces joined with the powerful tools of probabilistic calculus can be mixed together for a flexible, yet principled account of term dependencies for IR. In our model, dependencies are neither represented as additional dimensions, nor stochastically as joint probabilities. They assume a new status as *superposition* events. The relationship of such an event to the traditional term events are encoded by the off-diagonal values in the corresponding projection matrix. Both documents and queries are associated to density matrices estimated through the maximization of a product, which in the classical case reduces to a likelihood. As our model integrates the dependencies in the estimation phase, it has no need for combination parameters. Experiments showed that it performs equivalently to the existing dependence models on newswire test collections and outperforms the latter on web data.

To our knowledge, this work provides the first experimental result showing the usefulness of this kind of probabilistic calculus for IR. The marriage between vector spaces and probability can be endlessly improved in the future. One straightforward direction is to relax the assumption that single terms represent orthogonal projectors. This could lead to a new way of integrating latent directions as estimated by purely geometric methods such as Latent Semantic Indexing (LSI) [7] into a probabilistic model. In this work, we did not exploit the full machinery of complex vector spaces. We do not have a practical justification for the use of the complex field for IR tasks. However, we speculate that this could bring improved representational power and thus remains an interesting direction to explore. At last, we believe that our model could be potentially applied to other fields of natural language processing only by means of a principled Bayesian calculus capable of manipulating density matrices. We hope that this work will foster future research in this direction.

6. ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their valuable comments and suggestions.

7. REFERENCES

[1] J. Bai, Y. Chang, H. Cui, Z. Zheng, G. Sun, and X. Li. Investigation of partial query proximity in web search. In *Proc. of WWW*, pages 1183–1184, 2008.

[2] M. Bendersky and W. B. Croft. Modeling higher-order term dependencies in information retrieval using query hypergraphs. In *Proc. of SIGIR*, pages 941–950, 2012.

[3] M. Bendersky, D. Metzler, and W. B. Croft. Parametrized concept weighting in verbose queries. In *Proc. of SIGIR*, pages 605–614, 2011.

[4] R. Blume-Kohout. Hedged maximum likelihood estimation. *Phys. Rev. Lett.*, 105:200504, 2010.

[5] R. Blume-Kohout. Optimal, reliable estimation of quantum states. *New J. Phys.*, 12:043034, 2010.

[6] O. Chapelle, D. Metzler, Y. Zhang, P. Grinspan. Expected reciprocal rank for graded relevance In *Proc. of CIKM*, 2009.

[7] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *JASIST*, 41:391–407, 1990.

[8] J. L. Fagan. Automatic phrase indexing for document retrieval. In *Proc. of SIGIR*, pages 91–101, 1987.

[9] J. Gao, J. Y. Nie, G. Wu, and G. Cao. Dependence language model for information retrieval. In *Proc. of SIGIR*, pages 170–177, 2004.

[10] A. Gleason. Measures on the closed subspaces of a hilbert space. *Journ. Math. Mech.*, 6:885–893, 1957.

[11] K. S. Jones, S. Walker, and S. E. Robertson. A probabilistic model of information retrieval: development and comparative experiments. *Inf. Proc. Manag.*, pages 779–840, 2000.

[12] M. Lease. An improved markov random field model for supporting verbose queries. In *Proc. of SIGIR*, pages 476–483, 2009.

[13] C. Lee, G. G. Lee, and M. G. Jang. Dependency structure applied to language modeling for information retrieval. *ETRI*, 28(3):337–346, 2006.

[14] Y. Lv and C. Zhai. Positional language models for information retrieval. In *Proc. of SIGIR*, pages 299–306, 2009.

[15] A. I. Lvovsky. Iterative maximum-likelihood reconstruction in quantum homodyne tomography. *Journ. Opt. B6*, pages S556–S559, 2004.

[16] M. Melucci. Deriving a quantum information retrieval basis. *The Computer Journal*, 2012.

[17] M. Melucci and K. Rijsbergen. Quantum mechanics and information retrieval. *Advanced Topics in Information Retrieval*, 33:125–155, 2011.

[18] D. Metzler and W. Bruce Croft. Linear feature-based models for information retrieval. *Inf. Retr.*, 10(3):257–274, 2007.

[19] D. Metzler and W. B. Croft. A markov random field model for term dependencies. In *Proc. of SIGIR*, pages 472–479, 2005.

[20] D. Metzler, T. Strohman, Y. Zhou, and W. B. Croft. Indri at TREC 2005: Terabyte Track. In *Proc. of TREC*, 2005.

[21] M. Mitra, C. Buckley, A. Singhal, and C. Cardie. An analysis of statistical and syntactic phrases. In *Proc of RIAO*, pages 200–217, 1997.

[22] W. Morgan, W. Greiff, and J. Henderson. Direct maximization of average precision by hill-climbing, with a comparison to a maximum entropy approach. In *Proc. of HLT-NAACL*, pages 93–96, 2004.

[23] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2004.

[24] J. H. Park, W. B. Croft, and D. A. Smith. A quasi-synchronous dependence model for information retrieval. In *Proc. of CIKM*, pages 17–26, 2011.

[25] B. Piwowarski, I. Frommholz, M. Lalmas, and K. van Rijsbergen. What can quantum theory bring to information retrieval. In *Proc. of CIKM*, pages 59–68, 2010.

[26] M. Pretti. A message-passing algorithm with damping. *J. Stat. Mech.*, page P11008, 2005.

[27] J. Řeháček, Z. Hradil, E. Knill, A. I. Lvovsky. Diluted maximum-likelihood algorithm for quantum tomography. *Phys. Rev. A*, 75:042108, 2007.

[28] G. Salton, C. S. Yang, and C. T. Yu. A Theory of Term Importance in Automatic Text Analysis. *JASIST*, 26(1):33–44, 1975.

[29] M. D. Smucker, J. Allan, and B. Carterette. A comparison of statistical significance tests for information retrieval evaluation. In *Proc. of CIKM*, pages 623–632, 2007.

[30] F. Song and W. B. Croft. A general language model for information retrieval. In *Proc. of SIGIR*, pages 279–280, 1999.

[31] M. Srikanth and R. Srihari. Biterm language models for document retrieval. In *Proc. of SIGIR*, pages 425–426, 2002.

[32] H. Umegaki. Conditional expectation in an operator algebra. *Kodai Mathematical Seminar Reports*, 14(2):59–85, 1962.

[33] K. van Rijsbergen. *The Geometry of Information Retrieval*. Cambridge University Press, 2004.

[34] M. K. Warmuth and D. Kuzmin. Bayesian generalized probability calculus for density matrices. *Machine Learning*, 78(1-2):63–101, 2009.

[35] C. Zhai. Statistical language models for information retrieval a critical review. *Found. Trends Inf. Retr.*, 2(3):137–213, 2008.

[36] J. Zhao and Y. Yun. A proximity language model for information retrieval. In *Proc. of SIGIR*, pages 291–298, 2009.

[37] X. Zhao, P. Zhang, D. Song, and Y. Hou. A novel re-ranking approach inspired by quantum measurement. In *Proc. of ECIR*, pages 721–724, 2011.

[38] G. Zuccon and L. Azzopardi. Using the quantum probability ranking principle to rank interdependent documents. In *Proc. of ECIR*, page 357–369, 2010.