Prologue

Where did that come from?
Strictly Ballroom, film, directed by Baz Luhrmann, Australia: M&A Film Corporation, 1992.

Scene

A sunny office overlooking a cityscape of Victorian roofs and elm trees. K, an academic of some seniority judging by his white beard, and the capaciousness of his bookshelves, is sitting at his desk. The sign outside his door reads 'Please disturb'.

B: (A younger academic) enters without knocking, shortly followed by N (not so young).

B: I hear that you have been re-inventing IR.

K: Well, I am writing a book.

B: Yes, the story is that you have been looking at quantum mechanics, in order to specify a new model. Also (looks at N) that you are looking at quantum computation.

K: I have certainly been looking at quantum mechanics, but *not* because I want to specify a new model; I am looking at quantum mechanics because it gives insight into how one might combine probability, logic and vector spaces into one formalism. The role of quantum computation in all this is not clear yet. It may be that having reformulated IR in this way, using the language of quantum mechanics, that it will be obvious how quantum computation may help at the algorithmic level, but I have not been thinking that far . . .

N: (Interrupting) Well, I listen patiently as ever – but it seems to me that you are – yet again – taking an entirely system-based approach to IR

leaving no room for the user. For years now I have been saying that we need to spend more time on improving the interaction of the user with *any* system. Support for the user will make a bigger difference than any marginal improvements to a system. A new . . .

K: (Interrupting in turn) I know you think we should stop developing new theories and models and instead spend the time making existing ones work from a user perspective. Well, in a way that is what all this is about. Currently, we really do not have a way of describing formally, or in theoretical terms, how a user interacts with an IR system. I think . . .

N: – here we go. It has to be 'formal' –

K: we need a new paradigm, and the QM paradigm –

N: (Interrupting for the third time) Why? Why do we need this extra formalism? We have spent years describing how a user interacts with an IR system.

K: (Holds up hand) Hang on. We have had this argument over and over again. My reply has always been that if you do not formally describe or specify something then trying to arrive at a computational form becomes nigh impossible. Or if you do achieve a computational form without formal description then transferring a design from one approach or system to another becomes a nightmare. There is also the scientific imperative, that we cannot hope to make predictions about systems if we cannot reason about their underlying structure, and for this we need some kind of formality, and, dare I say it, –

N: I suppose I can't stop you –

K: a theory. Einstein always claimed that you need a theory to tell you what to measure.

N: Must you drag Einstein into this?

B: Let me get a word in edgewise. One could argue that a computer programme is a description, or a formal theory of a system. Why do we need more than that?

K: (Becomes instantly enthusiastic) Good question. It is certainly true that a computer program can be considered as a formal description of a process or a theory. Unfortunately it is very difficult to reason about such a description, and it is difficult to recover the semantics. What's more, computer programs are strongly influenced by the design of the digital computer which they run, that is, their von Neumann architecture. In developing this new IR paradigm I intend it perhaps to be implemented on a quantum computer.

N: Delusions of grandeur. So, tell us what is the essence or central idea of your new way of looking at things?

K: (Becomes even more enthusiastic) This will take some time, how long have you got?

B, N: We have got all afternoon.

K: (Hesitates) Of course, it would easier for you to understand what I am doing if you knew some elementary quantum mechanics. Let's see: you could start with Hughes' book on 'The Structure and Interpretation of Quantum Mechanics' . . .

N: I said we had this afternoon, not the next five years.

K: . . . I found his account invaluable to understanding some of the basics.

B: Can't you just give us the gist?

K: (Gets up and inspects his bookshelf) Well, the story really begins with von Neumann. As you know, in the thirties he wrote a now famous book on the foundations of quantum mechanics. One could argue that all later developments in quantum logic and probability are footnotes to his book. Of course von Neumann did not *do* QM, like say Feynman and Dirac, he theorised about it. He took the pioneering work of Bohr, Schrödinger, Heisenberg, Born and others, and tried to construct a consistent formal theory for QM It is much in the same spirit as what I am attempting for IR.

N: (Laughs) When I ascribed you delusions of grandeur I underestimated you. Are you now equating QM and IR in importance? Or merely yourself with von Neumann? In IR we deal only with artefacts and the way humans interact with them. Everything is man made. Whereas in QM we attempt to describe a piece of reality and many of the paradoxes arise because we are uncertain how to go about that.

K: (Focusing on the last point) Ah, exactly. You have put your finger on the problem. Both in IR and QM we are uncertain about how to describe things – be they real or artificial. In QM we have the problem of measurement; we don't know how to model the result of an observation which arises from the interaction of an 'observable' with a piece of reality. In IR we face the same problem when we attempt to model the interaction of a 'user' with an artefact.

B: (Gloomily) This is all getting a bit abstract for me. How about you try to make it more concrete?

K: (Cheerfully now) Well imagine the world in IR *before* keywords or index terms. A document, then, was not simply a set of words, it was much more: it was a set of ideas, a set of concepts, a story, etc., in other words a very abstract object. It is an accident of history that a representation of a document is so directly related to the text in it. If IR had started with documents that were images then such a dictionary

kind of representation would not have arisen immediately. So let us begin by leaving the representation of a document unspecified. That does not mean that there will be none, it simply means it will not be defined in advance.

B: (Even gloomier) Great. So how do I get a computer to manipulate it – this piece of fiction?

K: Actually that is exactly what it is – a document is a kind of fictive object. Strangely enough Schrödinger . . .

N: (As an aside) Here we go with the name dropping again.

K: (continues, ignoring N) . . . in his conception of the state-vector for QM envisaged it in the same way. He thought of the state-vector as an object encapsulating all the possible results of potential measurements. Let me quote: 'It (ψ-function) is now the means for predicting probability of measurement results. In it is embodied the momentarily attained sum of theoretically based future expectation, somewhat as laid down in a catalogue.'[1] Thus a state-vector representing a document may be viewed the same way – it is an object that encapsulates the answers to all possible queries.

N: (Perks up) Ah, I can relate to this. You mean a document is defined with respect to the queries that a user might ask of it?

K: Yes, in more than one way, as will emerge later. By the way, one could view Maron and Kuhns' original paper on probabilistic indexing in this sort of way. Indeed, Donald Mackay (1969, 1950), who worked with Maron, anticipated the use of QM in theorising about IR.

N: Good, keep going; we seem to be getting somewhere at last.

K: So what have we got? We have a collection of artefacts each of which is represented by a highly abstract object called a 'state-vector'. Of course using the term 'vector' gives the game away a little. These abstract objects are going to live in some kind of space (an information space), and it will come as no surprise to you that it will be a vector space, an infinite-dimensional vector space: a Hilbert space.

B: (With some frustration) Terrific. After all this verbiage we end up with a vector space, which is a traditional IR model. So, apart from being able to add ourselves as footnotes to von Neumann, what is the big deal?

K: The big deal is that we do not say in advance what the vectors in this space look like. All we require is a notion of dimensionality, which can be infinite, and objects that satisfy the axioms of a vector space, for example, vectors can be added and multiplied by scalars. Moreover, the

[1] Schrödinger, p. 158 in Wheeler and Zurek (1983).

space has a geometry given by an inner product which allows one to define a distance on the space. The fact that it is infinite is not immediately important, but there is no reason to restrict the dimensionality.

B: Why do you talk of scalars and not of real numbers?

K: You noticed that did you? Well, scalars here can be complex numbers.

N: Hold it, are you saying that we can attach a meaning to complex or for that matter imaginary numbers in IR?

K: No, I am not saying that. I am implying that we do not need to restrict our representational power to just real numbers. Rest assured that our observation or measurements will always deliver a real number, but it may be that we represent things on the way by complex numbers. There are many examples in mathematics where this is done, in addition to quantum mechanics, for example, Fourier analysis.

B: I don't buy this. Why introduce what appears to be an unnecessary complexity into the representation? What on earth would you want to represent with complex numbers?

K: To be honest I am not sure of this yet. But a simple example would arise in standard text retrieval where both term-frequency and document-frequency counts are used (per term, or per dimension) during a matching process. I imagine that we may wish to represent that combination of features in such a way that algebraic operations on them become easier. Right now when we combine *tf* and *idf* their identities are lost at the moment of combination.

N: So, from a mathematical, or algorithmic, point of view this may make sense. But, tell me, are you expecting the user to formulate their queries using complex numbers? If so, you can forget it.

K: No, of course not. But just as a person may write down a polynomial with real coefficients which has *complex roots*, a user may write down a query which from another point of view may end up being represented by complex numbers. The user is only expected to generate the point of view, and in changing it the query will change.

N: (With some impatience) This sounds great but I do not fully understand it. What do you mean by a 'point of view'?

B: Yes, what do you mean? I am lost now.

K: In conventional index term based retrieval the point of view in the vector space model is given by the axes in the space corresponding to the index terms in the query. Thus, if the query is $(a, b, c, \ldots)$ then a might lie along the x-axis, b the y-axis, c the z-axis, etc. Usually these are assumed to be orthogonal and linearly independent. Notice how convenient it is that the user has specified a set of axes. Now imagine that the query is

simply an abstract vector in the space, we would still have to define it with respect to the basis of the space, but it would be up to us, or the user, to refer the objects in the space to different bases depending on their point of view. A change of basis constitutes a change of point of view.

B: Well, I am not sure this buys us anything but I'll hang in there for the moment. I see that you are still talking about queries as vectors. I infer that much of what you have said so far is a dressed up version of the standard vector space model of IR. Am I right?

K: You are right. I am trying to inspire the introduction of some of the new ways of talking by referring to the old way.

N: Get on with it – I am still waiting too.

K: All right. But first here is a small example of how we can go beyond standard vector space ideology. By assuming that the query is a vector in a high (maybe infinite) dimensional space, we are making assumptions about the dimensions that are not mentioned in the query. We could assume that those components are zero, or have some other default value. Why? No good reason, and perhaps the query would be better represented by a *subspace*, the subspace spanned by the basis vectors that are mentioned in the query. So we have grasped the need for talking about subspaces. The problem is how to handle that symbolically. More about this later.

(B and N look bored, so K quickly moves on)

K: Given the space of objects is a Hilbert space which we may fondly call an information space. How do we interact with it?

N: (With a sigh of relief) At last something about interaction.

B: Shut up, N. Let him talk. Although, I am still puzzled about how you will interact with these objects when you do not describe them explicitly in any way.

K: (With a grin) That is right. I forgot to tell you that. Once you have specified the basis (point of view) for the space, you can express the object in terms of the basis. This is done by projecting the object onto the different basis vectors. The effect of this is to give a 'co-ordinate' for the object with respect to each basis vector. It is a bit like defining an object by giving the answers to a set of simple questions, one question for each basis vector. If the object (state-vector) is normalised these projections are given by calculating the inner product between each basis vector and the state-vector. Of course, if we allow complex numbers then we would need to take the modulus (size) of the inner product to get a real number. In the case where we have a real Hilbert space, the state-vector

is simply expanded as a real linear combination of the basis vectors. The expansion would differ from basis to basis.

N: You are getting too technical again; let's get back to the issue of interaction.

B: Yes, let's.

K: The basic idea is that an observable, such as a query or a single term, is to be represented by a linear operator which is self-adjoint in the Hilbert space. This means that in the finite case it corresponds to a matrix which can have complex numbers as entries but is such that the conjugate transpose is equal to itself. Let me illustrate. If A represents an observable, then A is self-adjoint if $A = A^*$.
(K writes some symbols on the white board)

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

$$A^* = \overline{A'} = \begin{pmatrix} \overline{a} & \overline{c} \\ \overline{b} & \overline{d} \end{pmatrix} = A$$

$$\Rightarrow a = \overline{a},\ d = \overline{d} \text{ and hence real,}$$

$$\text{also } b = \overline{c},\ \overline{b} = c.$$

An example is

$$A = \begin{pmatrix} 1 & -i \\ i & 2 \end{pmatrix}$$

$$A^* = \overline{\begin{pmatrix} 1 & -i \\ i & 2 \end{pmatrix}'} = \begin{pmatrix} 1 & -i \\ i & 2 \end{pmatrix} = \mathbf{A}.$$

K: I know what you are going to say, what has this got to do with queries and users?

N, B: How did you guess, so what has it got to do with them?

K: Bear with me a little longer. The notion of representation is a little indirect here. In quantum mechanics the idea is that the value of an observable is given by the eigenvalues of the matrix.[2] The beauty is that the eigenvalues of a self-adjoint matrix are always *real*, even though the entries in the matrix may be complex. So here we come back to the fact that our representation may involve complex numbers but when we make a measurement, that is interact, we only get real results.

B: Hang on a bit, you said that the value of an observable is an eigenvalue, any eigenvalue? So, how do I know which one? Let me take a simple

[2] More correctly, this should say that the outcome of a measurement of the observable is given by an eigenvalue. See Appendix II.

example, when the observable has just two values, 1 and 0. How do I know which? Is this the right question to ask?

K: We are now getting to the meat of it. If your observable represents a two-valued question, '1' means 'yes' and '0' means 'no', then determining which answer is a matter of *probability*. For example, if your observable was to determine whether an object was about the concept 'house', then there would be two eigenvalues, one corresponding to 'house' and one corresponding to 'not-house'. The probability of each of these answers would be derived from the *geometry* of the space.

N: You have lost me . . . again. Where do the concepts 'house' and 'not-house' come from? One minute we have an observable which corresponds to a query about 'houseness', next we have concepts, presumably represented in the space, how?

K: Yes, that is right. I need to tell you about the idea of *eigenvectors*.

B: (With some despair) Oh no, not more algebra, is there no end to it?

K: (Soothingly) We are almost there. Corresponding to each eigenvalue is an eigenvector. So, for a self-adjoint operator (that is, an observable) you get a number of eigenvectors corresponding to the concepts underlying the observable. It so happens that these eigenvectors make up a basis for the space and so generate a point of view.[3] It is as if we have found a set of concepts, one corresponding to each eigenvector, with respect to which we can observe each document in the space.

B: What about this relationship between probability and the geometry of the space?

K: I will come to that in a minute.

N: (Somewhat grimly) I am glad to hear it, these algebraic considerations are starting to give me a headache. I thought all this was for IR? Anyway, proceed.

K: For the simple case where the observable represents a Yes/No question, the linear operator is a particularly simple, and important one: a projection operator. It is a theorem in linear algebra that any self-adjoint linear operator can be resolved into a linear combination of projection operators. In other words, any observable can be resolved in to a combination of yes/no questions. Although a projector may be represented by a matrix in n dimensions, it only has two eigenvalues. In general you would expect an n-dimensional matrix to have n eigenvalues.

[3] There is an issue of 'degeneracy': when an eigenspace corresponds to an eigenvalue, its dimension is equal to the degeneracy of the eigenvalue.

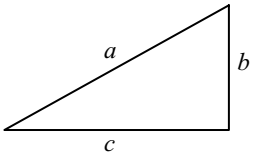
Projectors have two. The effect of this is that there is a certain amount of degeneracy, which means that corresponding to each eigenvalue we have an eigenspace, and together these two eigenspaces span the entire space.

B: What about the basis? If the space is n-dimensional, we need n basis vectors to make up the basis.

K: That is still so, except that within each subspace you can choose an arbitrary set of basis vectors spanning the *sub*space. Adding these two sets will give a set of basis vectors spanning the whole space. This finishes the geometry.

N: (Deliberately obtuse) What geometry? I only see vectors, subspaces, bases, and operators. Where is the geometry?

K: You are right to be suspicious, the geometry is implied, and it is used to give us both a logic and a probability measure. To calculate the probability of a particular eigenvalue we project orthogonally the state-vector down onto its eigenspace and measure the size of that projection in some way to get the probability. Probability measures have to satisfy some simple constraints, like for example that the sum of the measures of mutually orthogonal subspaces, that together exhaust the space, must sum to one. The geometry of the space through Pythagoras' Theorem ensures that this is indeed the case. Remember that theorem – (K quickly sketches it)



$$a^2 = b^2 + c^2$$

B: So a^2 has the value 1, where b^2 and c^2 are the measures of the corresponding subspaces. You slipped in the idea of probability rather neatly, but why should I accept that way of calculating probability as being useful, or meaningful? You seem to be simply replacing the inner product calculation with a probability. Why?

K: A good question and a hard one. First let me emphasise that we use 'probability' because we find it intuitive to talk of the probability that an object has a certain property, or that is about something. Of course, in quantum mechanics this is shorthand for saying that if one attempted to measure such a property or aboutness then a result would be returned with a probability, possibly with a probability of one or zero. The problem is how to connect that probability with the geometric structure of

the space in which the objects reside. I will need to develop the abstract view a little further before I can totally convince you that this is worth doing.

N: Oh, no, not more mathematics.

B: Perhaps you can give us little more intuition about how to make this connection between the geometry and probability.

K: OK. But for further details I will have to refer you to a paper by William Wootters (1980a) and one by R. A. Fisher (1922), who were the first to moot the intuition I am about to describe. In fact Wootters developed a simple example in a very different context, which I will follow transposed to an IR context. But first let me go back to the pioneering work of Maron. Remember he developed a theory of probabilistic indexing in the sixties.

N: Yes, so he did, but as a model it never really took off, although the way of thinking in those early papers was very influential.

K: I agree, and it will serve here to interpret how the probability arises out of the geometry. Imagine that a document is designed (by the author, artist, photographer, . . .) to transmit the information that it is about a certain concept. One way to ascertain this information is to ask a large set of users to judge whether it is about that concept or not. A specific user answers either yes (Y) or no (N). Thus a long sequence, YNNYNY . . . , is obtained. We have assumed that our document is represented by a vector in a space, and that a concept is represented by a basis vector in the same space, the eigenvector of the observable representing the concept.[4] And so, geometrically, the extent to which that document is about the concept in question is given by the angle θ the document vector makes with the concept vector. We assume (following Wootters) that we are able to ask the users indefinitely, and that we cannot use the order in which the answers occur. You will agree that the probability, P, that a document is about the concept is given by the frequency of the Ys in the limit of the sequence, the size of sequence must not play a role. Now it turns out that the function $P = \cos^2\theta$ is the best code for transmitting a Y or N in the sense of maximising information that will tell us what θ is. One could describe this as a *content hypothesis*: 'The optimal way of displaying the content of a document in a vector space is to define the probability of a concept as the square of the modulus of projection of the state-vector on the concept vector'. This is a little

[4] The idea of representing documents and concepts in the same space is not new, Deerwester *et al.* (1990) discussed this at some length.