Information retrieval performance measures for a current awareness report composition aid

Thomas Krichel

Palmer School of Library and Information Science
College of Information and Computer Science
CW Post Campus of Long Island University
720 Northern Boulevard
Brookville NY 11548–1300, U.S.A.

Faculty of Information Technology
Novosibirsk State University
2, Pirogova Street
630090 Novosibirsk, Russia

http://openlib.org/home/krichel
krichel@openlib.org

Abstract

This papers studies a special "small" information retrieval problem where user satisfaction only depends on the ordering of documents. We look for a retrieval performance measure applicable for this setting. We define some requirements for such a measure. We develop a theoretical ordering of all outcomes. We look at some standard and purpose-build measures and assess them against the requirements. We conclude that a linear combination of two such measures is adequate.

1 Introduction

The classic measures of information retrieval performance are precision and recall. Precision is the number of retrieved and relevant documents divided

[⋆] I am grateful for comments by Christian Calmès, William S. Cooper, Robert M. Losee, Amanda Z. Xu and two referees of "Information Processing and Management". I am also grateful to all the volunteers of NEP for the work they have been putting into running the service. The hospitality of Tatyana I. Yakovleva provided a congenial setting for the work on this paper.

by the number of retrieved documents. Recall is the number of retrieved and relevant documents divided by the number of relevant documents. Both ratios are used jointly because they capture two complementary aspects of the retrieval process. Precision tells us how good the system is at filtering out non-relevant documents. Recall tells us how good the system is at finding relevant documents.

Other measures have been proposed that aim to summarize information retrieval performance in a single number. These include the average precision at seen document, the R-precision, the E-measure, van Rijsbergen's F and the average precision over all documents. All of these numbers are directly based on the concepts of precision and recall. In fact, they are precise mathematical functions of precision and recall ratios.

Despite numerous critiques of these measures, they remain the most widely deployed in "large" information retrieval problems. In such "large" problems there is a large set of documents, typically so large that the one-by-one examination of each document is not realistic. Then, an information retrieval system has the task of retrieving a set of documents that corresponds to the information need. The user only sees the set of retrieved documents.

This paper is motivated by my concern for a special information retrieval problem. We can call this problem a "small" information retrieval problem. The service helps a user to find the relevant documents out of a small collection of documents. The collection is small enough that the user can examine each document one by one. The purpose of the information retrieval system is to make it easier for users to reach decisions.

One example of such a "small" information retrieval system has been the motivation for this paper. I created the "NEP: New Economics Papers" service at http://nep.repec.org and I am involved in running it. NEP is a current awareness service of the RePEc digital library, see http://repec.org. It filters new additions to RePEc into weekly subject-specific reports. Each report is edited by a volunteer editor. Each week editors are given a list of new additions to RePEc. From that list, they select the documents that are relevant to the subject of the report. These "relevant" documents form an issue of the report. Report issues are circulated via email.

The NEP service has been running since 1998. During that time RePEc has grown. So has the list of new additions that appear each week. The median number of new documents per week, over the entire life of the service, is 300. But in recent months, bumper crops of over 600 new documents are not uncommon. With that sort of numbers, we can not require volunteer editors to ponder over each single one for much more than a few seconds. At present, most editors, when composing the report issue, look first at document titles.

If a title looks appealing, they may look at the abstract. But sometimes a non-appealing title may hide a relevant document. This would become clear if the editor has read the abstract. However, with large new additions lists it is not realistic to expect the editor to read the entire set of abstracts. A pre-selection by the title is inevitable. It inevitably leads to editorial mistakes.

To make life easier for the editors, it would be useful to sort the list of new additions in order to show editors up-front those documents that are most likely to be included. This system can bring two benefits to editors. First they do not have to labor through the whole of the new additions list. If the algorithm works well, they can skip the tail end. Second, they can increase attention to the documents that the computer has put to the top of the list. This has the potential to eliminate oversight of documents with imaginative titles.

To evaluate such a sorting system, we need some measures. Within the specific context, precision and recall do not appear to be useful. There are three basic ways in which one can interpret precision and recall within the NEP context.

One approach is to say that they remain constant. Recall is always 100%, and precision is always equal to the number of documents that are relevant to the subject report, divided by the size of the new additions list. If we take that view, then precision and recall do not depend on the sorting process.

Another approach looks at precision and recall at the level of the last relevant document. Thus, if the information retrieval system sorts all the relevant documents to the front, then precision and recall are 100%. If there are some non-relevant documents that are found before the last document that is relevant, precision is the number of relevant documents divided by the position of the last relevant document. But recall is still 100%, therefore it is not useful. The measure of precision that we reach with this view is isomorph to the Nosel measure that I discuss in Subsection 4.4.

A third approach is to think of the output of the information retrieval system as sets of documents. We distinguish the documents that the information retrieval system has predicted as relevant versus those that it has predicted as non-relevant. We can then calculate precision and recall figures as intended by comparing the sets returned by the information retrieval system with the sets assembled by the editor. The latter are assumed to be correct. There are two problems with this approach. First, all computer-based information retrieval systems rank documents by the likelihood that they are relevant. It is in the very nature of the type of calculations done that such a ranking is produced. Therefore, if we conceive the information retrieval system output as a couple of sets, we exclude additional information that the system has produced. This is inconsistent with an assumption of rational behavior of users. Second, it should matter a lot in what order the documents appear in. This second problem is best illustrated by an example. For the sake of illustration, let numbers denote documents that, according to the editor, are relevant, and letters denote documents that are not relevant. Assume that the information retrieval system finds that documents 1 and 2 are relevant. It will sort those to the front. Then $[1, 2, 3, 4, \mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}]$ and $[1, 2, \mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}, 3, 4]$ have the same precision (100%) and recall figures (50%). But the former is perfect for the editor, while the latter is basically useless. Half of the relevant documents are at the front, the other half is at the rear. The editor has to still work through the entire list of new additions to find the all relevant documents.

We hope to have convinced the reader of the need for alternative measures. The remainder of the analysis therefore does not start with the analysis of precision and recall values, as Van Rijsbergen (1974) or Shaw (1986) have done. Instead we use an informal utility maximization approach, as, for example, Cooper (1973). In addition, we will consider rationality arguments, an approach that is more often found in economics rather than in information science. Before we review criteria, we devote two sections to further thinking about the problem. In Section 2 we set out the general framework. In Section 3 we look more closely at the problem that editors face. We try to establish what order a rational editor may place on the outcomes. In Section 4 we present alternative measures. In Section 5 we test these measures on the NEP report data using support vector machines. In Section 6 we offer conclusions.

2 The problem

Let there be a vector x called the outcome vector or simply the outcome. It has n elements, r of which take the value 1, i.e. they represent relevant documents, and $n - r$ take the value 0, i.e. they represent non-relevant documents. The ratio r/n is the generality of the information need. Let us call $\mathbb{O}(r, n)$ the outcome set. This is the set of all possible outcomes. For a fixed r and a fixed n, this is a set with $n!/(r!\,(n - r)!)$ members.

Loosely speaking, we are looking for a measure of how much the 1s are at the front of the vector and the 0s at the end. The best outcome is

$$x_{\mathrm{b}} = [1, \ldots, 1, 0, \ldots, 0]$$

and the worst outcome is

$$x_{\mathrm{w}} = [0, \ldots, 0, 1, \ldots, 1].$$

Let $f(x)$ denote a measure of the quality of the outcome. There are many measures that one may define. Each of these measures is subjective in a way.

It captures a desired property of the outcome. We can find four requirements that hopefully most people can agree with.

Requirement 0 is the most subjective. It says that $f(x)$ must not be too complicated. It should be fairly easy to explain to people what the measure is.

Requirement 1 is in some ways a corollary of requirement 0. People are used to reason that a better outcome means a higher $f(\cdot)$. Thus

$$f(x) > f(x') \quad \iff \quad x \text{ is better than } x'.$$

Requirement 2 again is a corollary of requirement 0. People are used to reasoning in term of percentages. Therefore it seems adequate to require

$$f(x_{\mathrm{b}}) = 1. \tag{1}$$

Apart from the best outcome another benchmark is important. That is the case where x is picked randomly out of $\mathbb{O}(r, n)$. In that case, the outcomes are of no use. This leads to Requirement 3

$$\mathrm{E}\, f(x) = 0, \tag{2}$$

where E stands for the expected value operator. One technical constraint that comes out of this requirement is that there has to be a closed form for the expected value. Of course, the expected value of any measure $f(x)$ over the finite set $\mathbb{O}(r, n)$ can be calculated by computer as the average over all potential outcomes. However, for large n and r, the number of members of $\mathbb{O}(r, n)$ becomes too large for this to be practical. Current computer technology is simply not powerful enough to accomplish the calculations in reasonable time.

Finally, we have an additional desired feature. We refer to this as the scaling property. With a constant generality r/n, if r and n are both multiplied by a scale $t \in \mathbb{N}$, with $t > 1$, we would like to get the same values of $f(\cdot)$ if we construct scaled outcome by repeating each element in the original vector t times. For an example, assume $n = 3$, $r = 1$ and $x = [0, 1, 0]$. If $t = 2$ we can construct $x' = [0, 0, 1, 1, 0, 0]$, by simply repeating each element in x t times. It is obvious that we can always make such a scaling transformation and that this transformation is unique. Let $t \otimes x$ denote the scaled outcome vector. If we make such a transformation, it appears natural to wish that $f(x)$ does not change, i.e.

$$\forall\, x \in \mathbb{O}(r, n), \quad \forall\, t \in \mathbb{N} : \qquad f(x) = f(t \otimes x).$$

If $f(\cdot)$ satisfies to that property, we will say that it scales.

3 Subject editor behavior modeling

The proposed system helps editors of a current awareness service such as NEP. The ultimate judge of performance is therefore the subject report editor. To build general criteria, it is useful to build a model of subject editor behavior. While a full mathematical model would be outside the scope of this paper, we hope to establish some general principles using simple deductive reasoning based on a highly simplified view of the editorial process.

An editor faces a list of documents. A document is metadata about a paper plus a link to the full text of that paper. An editor may spend a lot or only a little time on the document. This decision on the level of effort per document is difficult to model. Therefore we will not look at it here. In other words, we assume that the examination of a document is a discreet process i.e., the document is examined or it is not examined. Requirement 1 is in some ways a corollary of requirement 0. People are are used to reason that a better outcome mean a higher $f(\cdot)$.

After examining a document, the editor knows whether that document is relevant or not. We further assume that the decision to include a document or to exclude it only depends on the contents of that document. It is independent from the contents of other documents. This reasoning assumes away any learning that may take place while the list of documents is examined.

As the editor works through the list of documents, she faces two types of costs. First there is the cost c_1 of examining the next document. Without loosing much generality we can assume that c_1 remains constant over the report issue composition process. Second, there is the cost of missing relevant documents. Let us loosely call this c_2, though it is clear that c_2 somehow depends on actual number of documents missed. As the editor moves along the list, she faces an optimal stopping problem. If she stops to examine documents, she no longer suffers the penalty c_1 from examining all the following documents. But she faces the penalty c_2 of missing relevant papers. If $c_1 >> c_2$, the editor will not examine any documents at all, and if $c_2 >> c_1$ the editor will examine all documents. In less extreme cases, there is a balancing act. This balancing act is complicated because of the uncertainty surrounding c_2.

To make further progress in our reasoning, we need to simplify the problem. Let us assume that a magical interface could be built that would remove the uncertainty regarding c_2. We could imagine a traffic light sign in the editor's interface. It would show green while there is at least one more relevant document, and it would show red if there is no more relevant documents. Such a scenario is unrealistic of course, but for the moment just imagine it could be achieved. Clearly when the traffic light turns red, the editor will stop examining new documents. Now let us in addition assume that the editor is a conscientious person. By this we mean that while the traffic light is green, she will continue to examine new documents, until the traffic light is red.

However unrealistic this scenario of the traffic light scenario is, it can teach us one insight. With a traffic light, the editor will, when presented with two outcomes x and x' prefer the one where the last value of i where x_i is 1, say i^*, $i^* = \arg\max_i : x_i = 1$ has the lower value. For a conscientious editor examining either x or x', $c_2 = 0$. However the examination cost c_1 will be lower the lower i^* is. This reasoning establishes a weak ordering over all outcomes. Comparing two outcomes, an editor will prefer the one with the last relevant document at an earlier position. The editor will be indifferent between two outcomes that have the last relevant documents at the same position. This reasoning requires under certainty, and that the editor is conscientious. We can hope that it will also hold under some uncertainty, provided that the uncertainty is not too large, and for less then full conscientious editors, provided they are not reckless.

Now imagine that the traffic signal can not be completely trusted. It would get it most of the time right but not always. Let us assume, to simplify, that the uncertainty would only hold between the second-to-last and the last document. Assume that the editor would still follow the rule to stop if the traffic light turns to red, simply because the uncertainty is marginally small. Let i^* be the position of the last relevant document. Compare two outcomes that are identical, save the fact that positions $i^* - 1$ and $i * -2$ are exchanged

$$x_1 = [\ldots, 0, 1, 1, 0, \ldots, 0]$$
$$x_2 = [\ldots, 1, 0, 1, 0, \ldots, 0].$$

In the notation above $\ldots$ represent positions that are identical for both outcomes. The second 1 is the last, i.e. it is followed by zeros only.

I claim that when the editor compares the two outcomes, she will prefer x_2 over x_1. The reasoning goes as follows. If a wrong red signal is perceived at position $i^* - 1$, the editor looses the last relevant document under x_2, but two relevant documents under x_1. If a wrong signal is received at any earlier position than $i^* - 1$ the loss of the number of relevant documents is the same. Therefore, when presented with two outcomes that have the same position for the last relevant document, the editor will prefer the one with the second-to-last document at the earlier position. This reasoning can be repeated for the third-to-last document etc. We obtain a complete order over the outcomes. Let us call it the natural order.

We can conjecture that it is possible to find a class of functional specifications for the loss function, and a class of distribution functions of documents in the included/excluded domain such that, when editors minimize total loss knowing the distribution function, they prefer outcomes in natural order. Developing such a generic class would, however, go beyond the scope of this paper.

4 Various measures

In this section we address different measures for the literature or purpose-built for this paper.

4.1 The Swets/Brookes measure

One interesting measure was proposed by Swets (1963). He assumes that "when a search query is submitted to a retrieval system, the system assigns an index value (call it z) to each item in the store." This assumption holds true for all computerized text classification and information retrieval systems. Let $z|\mathrm{r}$ and $z|\mathrm{n}$ be the value that the system assigns to z given that it is relevant and non-relevant. Swets (1963) proposes to evaluate

$$\frac{\mathrm{E}(z|\mathrm{r}) - \mathrm{E}(z|\mathrm{n})}{\sigma^2(z|\mathrm{n})},$$

where σ^2 denotes the variance. According to Swets (1963), as long as both conditional distributions of z are normal, the measure has desirable properties. But Brookes (1968) suggests that a better measure would be

$$\frac{\mathrm{E}(z|\mathrm{r}) - \mathrm{E}(z|\mathrm{n})}{\sqrt{\sigma^2(z|\mathrm{r}) + \sigma^2(z|\mathrm{n})}}. \tag{3}$$

This measure truly expresses the discriminating power of the underlying information system. The z data that it uses is much richer than the positional data used by other measures proposed in the following here. On the other hand, its precise values are dependent on the technical characteristics of the information retrieval system. Therefore, its best use is as a technical tool to compare different parameters within the same information retrieval methodology.

An important problem is that the measure violates requirement two[1] The

[1] It is not trivial to normalize the value such that it is 1 in the best case. One approach is that, faced with an outcome x and its associated z vector, we can construct a perfect outcome that would rearrange x to have all relevant documents at the top, while keeping the z values constant. Unfortunately, while the constructed best outcome certainly has a better value in term of the numerator in the expression 3, we can not be sure about which way the denominator is going. Therefore we can not construct an artificial best outcome in this intuitive way.

Table 1
The Aselt measure for $r = 2$ and $n = 5$

x	$\alpha(x)$	$a(x)$
1,1,0,0,0	1.5	1.0
1,0,1,0,0	2.0	2/3
0,1,1,0,0	2.5	1/3
1,0,0,1,0	2.5	1/3
0,1,0,1,0	3.0	0
0,0,1,1,0	3.5	−1/3
1,0,0,0,1	3.0	0
0,1,0,0,1	3.5	−1/3
0,0,1,0,1	4.0	−2/3
0,0,0,1,1	4.5	−1.0

measure has one more problem. It is a measure that is essentially linear. The numerator is linear, and the sign of the expression only depends on the numerator. This causes a problem that we discuss in the next subsection.

4.2 The Aselt measure

Although Swets/Brookes measure is rarely used at present, linear measures have not disappeared. For example, Losee (1998) considers the "average search length" as a candidate measure for information retrieval performance. The average search length is the average position of units within the vector x. That is, it is the sum of the positions i where $x_i = 1$, divided by n. If $a(x)$ is the average search length, we have

$$\alpha(x_{\mathrm{b}}) = \frac{1+r}{2} \qquad \text{and} \qquad \alpha(x_{\mathrm{w}}) = \frac{2\,n - r + 1}{2}.$$

The expected value for the average search length[2] can be readily found as

$$\mathrm{E}\,\alpha(x) = \frac{n+1}{2},$$

which, interestingly, does not depend on r. Having found the expected value, we can construct a measure that satisfies requirements 2 and 3. We define the Aselt—an acronym for "average search length transformed"—measure of an

[2] This could be labelled the average average search length or expected average search length. Our use of acronyms avoids confusion.

Table 2
Lofop measure for $r = 2$ and $n = 5$

x	$m(x)$
1,1,0,0,0	100.00%
1,0,1,0,0	73.38%
0,1,1,0,0	52.73%
1,0,0,1,0	35.86%
0,1,0,1,0	15.22%
0,0,1,1,0	−11.40%
1,0,0,0,1	−28.27%
0,1,0,0,1	−48.92%
0,0,1,0,1	−75.54%
0,0,0,1,1	−113.06%

outcome x, $a(x)$, by

$$a(x) = \frac{n + 1 - 2\,\alpha(x)}{n - r}.$$

The Aselt measure is nicely bounded $a(x_{\mathrm{b}}) = 1$, $a(x_{\mathrm{w}}) = -1$. It satisfies requirements 1–3. And it scales, because it is based on an average, which itself is a linear function.

Table 1 carries a numeric illustration of the Aselt measure. Looking at it we have two remarks. First, there seems to be a concentration of points towards the middle of the distribution. Second, not every different outcome has a different Aselt measure. In particular, the measure does not penalize two relevant documents in the middle of the vector any different than one relevant document in the end. From a managerial point of view, this is a problem if we are much concerned about having a measure that penalizes heavily a single document that is left out at the end.

In Table 1 outcomes are sorted by the natural order. The table illustrates that the Aselt measure violates the natural order. That is, there are some couples of outcomes x and x', where x is better than x' according to the natural order but $a(x) < a(x')$. We refer to the non-respect of the natural order as "natural order reversion" in the following. It is a problem that is generic to all linear measures, and therefore generic to all scaling measures. It therefore also affects the Swets and Brookes measures.

4.3 Lofop measure

One idea to combat natural order reversion is to use the natural logarithm, in order to reduce the high numbers that occur with the findings of relevant documents at the top. If a document in position i is relevant, let it increment a total quality indicator of the outcome by $\ln i$. Let $\mu(x)$ be this measure. Let $x_i = 1$ if a relevant document is at position i, or 0 otherwise. We get

$$\mu(x) = \sum_{i=1}^{n} x_i \ln i,$$

which is well defined. A simple calculation shows that

$$\mathrm{E}\,\mu(x) = \ln(n!)\,\frac{r}{n}.$$

This motivates the definition of the Lofop—an acronym for <u>lo</u>garithm of <u>fo</u>und <u>p</u>osition"—measure $m(x)$ as

$$m(x) = \frac{\mu(x) - \mathrm{E}\,\mu(x)}{\mu(x_{\mathrm{b}}) - \mathrm{E}(\mu(x))}.$$

Note that the $\mu(x)$ could also be defined for the logarithm to another base. Normalization leaves the actual $m(x)$ unchanged for any base. Table 2 suggests that the Lofop measure does have desirable properties. It spreads outcome values more evenly than the Aselt measure and, according to the table, it obeys the natural order. Unfortunately, the respect for the natural order of the Lofop measure in the $r = 2$, $n = 5$ case is not a general rule. We don't have to look far before reversion on the natural order raises its ugly head again. We will leave it as an exercise for the reader to show that $m(1, 1, 0, 0, 0, 0, 0, 0, 1) = 2.64\%$ but that $m(0, 0, 1, 0, 0, 1, 1, 1, 0) = -15.64\%$. This is a clear violation of the natural order.

4.4 The Nosel measure

Cooper (1968) looked at a more general model than we do here. He assumed that the information system would result in a weak ordering of documents. That is, some documents would be ranked exactly as relevant as some others. In that situation, he assumes that the user will look at these equally ranked documents in a random order. He calls the measure that he proposes the "expected search length". It is the number of non-relevant documents that a user would find until she finds a target number of relevant documents. In our setting, his "expected search length" reduces to the search length, because there is no uncertainty about how the order in which the documents are examined.

Table 3
The Nosel measure for $r = 2$ and $n = 5$

x	$\lambda(x)$	$l(x)$
1,1,0,0,0	0	1
1,0,1,0,0	1	1/2
0,1,1,0,0	1	1/2
1,0,0,1,0	2	0
0,1,0,1,0	2	0
0,0,1,1,0	2	0
1,0,0,0,1	3	−1/2
/0,1,0,0,1	3	−1/2
0,0,1,0,1	3	−1/2
0,0,0,1,1	3	−1/2

A further simplification in our case is that we can consider that the target number of relevant documents is the true number of relevant documents r. Let λ denote the search length, then

$$\lambda(x_{\mathrm{b}}) = 0 \qquad \text{and} \qquad \lambda(x_{\mathrm{w}}) = n - r.$$

The expected value of the search length[3] over all outcomes in the outcome set is

$$\mathrm{E}\,\lambda(x) = \frac{(n-r)\,r}{r+1}.$$

The expected value is smaller than the worst case, but only a little bit smaller, especially if r is large. This suggests that the distribution of values is highly skewed.

In order to comply with requirements 1–3, we define the Nosel—an acronym for “normalized search length—measure of the outcome $l(x)$ as

$$l(x) = 1 - \frac{\lambda(x)\,(r+1)}{r\,(n-r)}.$$

As seen in Table 3, many different outcomes receive the same Nosel measure. The Nosel measure essentially expresses how low the last document found was. It is not interested in the position of any other document. This can be a strength of the measure, because it makes it easier to explain to people what has been measured. On the other hand it can also appear as weakness of the measure. While the position of the last document should be—according to our

[3] This is not the same thing as the expected search length. Our use of abstract geographical names avoids the potential confusion.

reasoning in Section 3—the most important aspect of editor satisfaction, it is doubtful that it should be the only one. In particular, an outcome that has all the relevant documents next to each other but away from the top should, be counted as worse than an outcome that has all the relevant documents at the top bar one at a late position, even if the position of the last relevant item is the same in both scenarios. This idea is, of course, embodied in the natural order.

The Nosel measure does not scale. To see this, it is sufficient to look at a counter example. $c(0,0,0,1,1) = -.5$, but $c(0,0,0,0,0,0,1,1,1,1) = -.25$, which is not even close.

4.5 The Ponori measure

Instead of looking at various measures and examining them if they fit the natural order, a more fruitful approach may be to build measures that directly impose the natural order by construction.

One idea is that we can consider the sequence of 1s and 0s in the outcome vector as a binary number. This binary number can be converted to decimal in order to capture its position in the natural order. Converting x_b to a decimal form leads to the highest number, and converting x_w to decimal leads to the lowest possible number. However, consider

$$x_\mathrm{b} = [1, \ldots, 1, 1, 0, \ldots, 0, 0]$$

versus

$$x' = [1, \ldots, 1, 0, 0, \ldots, 0, 1].$$

The difference in the decimal measure between x_1 and x_2 does not appear to be as significant as one would like, if one does wish to penalize late occuring relevant documents significantly. Therefore, rather than measuring the quality of the result by assigning high powers to the first outcomes, we invert the outcome vector. Thus we give high powers to the lower outcomes and call the resulting number a loss. Of course, we are not limited to considering powers of 2 as the binary-number interpretation suggests. Any power of $y > 1$ will be able to accomplish the purpose of implementing the natural order. This motivates the following definition.

Let $x = [x_1, \ldots, x_n] \in \mathbb{O}(r, n)$. Then the y-Ponori—an acronym for "$\underline{\text{po}}$lynomial $\underline{\text{n}}$atural $\underline{\text{or}}$der $\underline{\text{i}}$mposition"—penalty of x, $\omega(x, y)$ is

$$\omega(x, y) = \sum_{i=1}^{n} y^{i-1} x_i.$$

Table 4
Ponori measure for $r = 2$ and $n = 5$, $y = 2$ and $y = \infty$

x	$\omega(x, 2)$	$o(x, 2)$	$o(x, \infty)$
1,1,0,0,0	3	1.0	1
1,0,1,0,0	5	37/47	1
0,1,1,0,0	6	32/47	1
1,0,0,1,0	9	17/47	1
0,1,0,1,0	10	12/47	1
0,0,1,1,0	12	2/47	1
1,0,0,0,1	17	−23/47	−1.5
0,1,0,0,1	18	−28/47	−1.5
0,0,1,0,1	20	−38/47	−1.5
0,0,0,1,1	24	−58/47	−1.5

We find

$$\omega(x_{\mathrm{b}}, y) = \frac{y^r - 1}{y - 1} \qquad \text{and} \qquad \omega(x_{\mathrm{w}}, y) = y^{n-r}\, \omega(x_{\mathrm{b}}, y). \tag{4}$$

The expected value is

$$\mathrm{E}\, \omega(x) = \frac{y^n - 1}{y - 1} \frac{r}{n}. \tag{5}$$

We can substitute for (4) and (5) to obtain a measure that satisfies (1) and (2). This motivates the following definition. The Ponori measure of an outcome x at the power y is

$$o(x, y) = \frac{(y^n - 1)\, r - (y - 1)\, n\, \omega(x, y)}{(y^n - 1)\, r - n\, (y^r - 1)}.$$

As $y \to \infty$ the Ponori measure becomes as indicator if the last relevant document is at the last position. As $y \to 1$ $o(x, y) \to a(x)$. Thus, the Aselt measure is nothing but a limiting case of the Ponori measure. In this limiting case, the Ponori measure scales. But with $y > 1$ it does not scale. Table 4 illustrates the Ponori measure.

4.6 The Copnori measure

The dependency of the Ponori measure on y is inconvenient. It is not clear what y to choose. While the ordering of outcomes is not sensitive to the choice of any $y > 1$, the numbers coming out of the evaluation definitely are. Thus a more fundamental measure is called for.

Table 5
The Copnori measure for $r = 2$ and $n = 5$

x	$\kappa(x)$	$k(x)$
1,1,0,0,0	0	1.0
1,0,1,0,0	1	7/9
0,1,1,0,0	2	5/9
1,0,0,1,0	3	1/3
0,1,0,1,0	4	1/9
0,0,1,1,0	5	−1/9
1,0,0,0,1	6	−1/3
0,1,0,0,1	7	−5/9
0,0,1,0,1	8	−7/9
0,0,0,1,1	9	−1.0

One very simple way to achieve this is to count through the elements in the outcome set in the natural order, assigning each worse outcome an incremental penalty of 1. Computing the sequence is reasonably straightforward[4] If we start with counting at 0, we get the counts $\kappa(x)$ as

$$\kappa(x_{\mathrm{b}}) = 0 \qquad \text{and} \qquad \kappa(x_{\mathrm{w}}) = \frac{n!}{r!\,(n-r)!} - 1. \tag{6}$$

The expected value is readily found as

$$\mathrm{E}\,\kappa(x) = \frac{\kappa_{\mathrm{w}} + \kappa_{\mathrm{b}}}{2}. \tag{7}$$

We can substitute for (6) and (7) to obtain a measure that satisfies (1) and (2). This motivates a definition. The Copnori—an acronym for "constant penalty

[4] The detail of our computational implementation is as recursive. For any outcome vector, we first remove the trailing non-relevant outcomes. They will not affect the result. Thus we have a shortened outcome vector with n_1 elements, say, r of which are relevant. We can then calculate a minimum value for $\kappa(x)$ as $\kappa_{\mathrm{b}}(1) = (n_1 - 1)!/(n_1 - 1 - r)/r!$. We remove the last relevant outcome from the vector. This completes the first step. We have a new vector of $n_1 - 1$ element, $r - 1$ of which are relevant. Again, we remove any non-relevant outcomes of the end of that new vector. We find the next relevant outcome at n_2. We have a new minimum value, $\kappa_{\mathrm{b}}(2)$ which we add to the value found in the previous step $\kappa_{\mathrm{b}}(1)$, etc. We continue proceeding until we arrive at a vector that has only relevant outcomes. There we find the sum of all $\kappa_{\mathrm{b}}(\iota)$, where ι is the step number.

natural order imposition"—measure $k(x)$ of an outcome $x \in \mathbb{O}(r,n)$ is

$$k(x) = 1 - \frac{2\,(\kappa(x))}{\frac{n!}{r!\,(n-r)!} - 1}.$$

Unfortunately, the Copnori measure does not scale. This is seen with an example.

$$k(0,1,0,1,0) = 1/9, \qquad k(0,0,1,1,0,0,1,1,1,0,0) = 89/209.$$

Both numbers are not even close to each other. But, the measure has three strong points. First, if one understands the natural order, it is a very intuitive measure. Second, it does not depend on an arbitrary parameter. Third, given the algorithm that we developed, the Copnori measure is easy to compute even for large n and r. Table 4 illustrates the Copnori measure.

5 Test

Our aim is to develop a sorted list of new additions to RePEc for editors of NEP. In NEP each subject issue has a code nep-*xxx* where *xxx* is a sequence of three letters. A special report nep-all contains the list of all new additions to RePEc. Thus sorting the list of new addition is like sorting the nep-all report issue. Each time a new nep-all issue is produced, it is sorted for the use of the editors. For each subject report, the result of the sorting is different, of course[5]. To sort the nep-all for a subject report we look at the past subject report issue data. For each subject report, we have two sets. The first is the set of documents that have been included in the subject report. The second is the set of documents that have not been included in the report. The membership of the latter set is somewhat more difficult to determine than the former. Sometimes, an editor may not have looked at an entire list of new documents. This can happen, for example, if the editorship of a report is vacant. Therefore we restrict membership of the second set to all those documents in the nep-all issues for which at least one document of the nep-all issue has been included from. Thus documents in nep-all issues in which no document appeared in the subject issue have been ignored. While this may be an oversight of negative learning examples, there are still plenty of negative example left, because the generality of subject reports is small.

[5] In NEP documentation the term "pre-sorting" rather than "sorting" is used. In NEP "sorting" is a different process than pre-sorting. Sorting occurs when an subject issue is being produced. After an editor has discarded non-relevant documents, (s)he may decide to sort the documents in an order such as to put the most interesting document right to the top of the issue. This is an optional step of the process of creating a new subject issue.

We treat the occurrence of documents in different reports as independent events. Data in Barrueco Cruz et al. (2003) suggests that this is not the case. However, in a practical application, it would be cumbersome to rerank a nep-all report for a certain subject when it becomes known that the editor of another subject report has included that document in her report issue, based on that new information, because editors make their decisions independently from each other but typically within a short time frame after the nep-all report has been issued. Thus, by ignoring co-occurrence of documents altogether, we are working under realistic operating conditions.

We keep feature extraction very simple. From each document, we use the author names, title, abstract, classification codes, and the serial in which the paper has appeared. We concatenate the resulting string. We remove all punctuation, transliterate to lowercase and collapse whitespace. Each whitespace-separated component of the resulting string is a feature. For each document, we count the occurrence of the feature f as t_f. The weight of the feature f in the document, w_f is then given as

$$w_f = \frac{t_f}{\sum_{\forall f} t_f^2}.$$

Note that it is not necessary to take document frequency into account here, because to form the ranking, we use support vector machines (SVM). This technique goes back to Vapnik (1995). It is now a widely used text classification technique. Ginsparg et al. (2004) provide one example in a similar context to ours. The svm_light software of Joachims (1999) runs all the calculations. According to Krichel and Bakkalbasi (2005) the median nep-all report has 300 documents. Therefore we set aside 300 randomly selected documents for testing. The rest we use for training the SVM. We conduct at least 10 runs for each report. For some reports, where the generality is low, some selected testing dataset contains no relevant document. In that case, we repeat runs until at least one among the 300 randomly selected testing documents is relevant. The results are so bulky that we have confined them to an appendix A. The Aselt measure gives a reasonable range of results, and shows, by its numerical values, that the performance of the SVM is really quite good. But we have rejected the measure on theoretical grounds. The same holds for the Lofop measure. We still include them in Table A.1 for the sake of completeness.

While the Ponori measure has desirable theoretical properties, there is a bad problem in tests where the set of outcomes to be ordered is large, say more than 100. In that case, if $y > 1$, any practical outcome that has the ability to lift the last document to say before the last third or last quarter of all the documents will get a measure that is close to 1, or even equal to 1 after rounding. As we increase the value of y, we are converging toward a situation where the outcome is 100.00% as soon as the last document is not relevant, and a negative number if it is. This clearly is not what we want.

A similar problem affects the Copnori measure. Recall that the Copnori measure gives each outcome its own entire number. Outcomes where a relevant document appears late are penalized very heavily. Therefore, as soon as the retrieval system is to able lift up all relevant documents from low positions, it does very well. Even if the last relevant document is in the middle, the measure shows a result that is close to 100%. In fact, is does so even more than the Ponori measure. The alert reader will note that there are a number of maxima in the table are 100% for the Copnori measure but less than 100% for the others. In the theoretical framework that we use, 100% is the value reserved for the optimal outcome x_{b}. It is therefore impossible for the same outcome to be evaluated 100% by one measure and less than 100% by another measure. The explanation for this apparent error is the table is rounding. The computer says its 100% when in fact is can not see any more the real value that is a tiny bit below 100%.

The Nosel measure somehow has the opposite problem. It only looks at the last position of the last document, it takes no account of the ability of the information retrieval system to put relevant documents to the front. However as we noted in the introduction, the feature is also important, because it allows the editor to spend extra efforts on the documents, reading more than the title.

Thus we can say that in principle, the Nosel measure "underestimates" the success of the system, whereas the Copnori measure "overestimates" the success of the system. However, this general statement only holds when the system is a success. When the result is lousy, Copnori exaggerates the bad performance. This is simple the reverse of the fact that the Copnori measure gives very good results for a large span of top-measures. Since both Copnori and Nosel have an expected value of zero at the random results, the Copnori compensates with more lower values at the tail end. Related to this, the variance of the Copnori measure is higher than the variance of other measures. Generally, when the results are quite good, the Nosel measure has the higher variance. This comes as no surprise since it only looks at one single element of the outcome vector, the one that comes lowest.

6 Conclusions

We think of precision and recall as set-based measures. Indeed, they are based on the idea that the total set of documents contains two complementary subsets, the subset of relevant documents and the subset of non-relevant documents. A query creates two other complementary subsets, the subset of retrieved and the subset of non-retrieved documents.

In this paper, we discuss a different class of information retrieval performance measures we call vector-based measures. Vector-based measures start with a different way of thinking about what is happening at query time. We think of the set of documents as a vector. Indeed, from a computational point of view, it is a vector, because all documents are in some order in the information retrieval system. The task of the information retrieval system is to sort all the documents that are relevant to the beginning of the vector, and sort the non-relevant documents to the end of the vector.

From our setup we have a theoretical ordering of outcomes we call the natural order. Therefore to evaluate the information system, we prefer measures that respect the natural order. It turns out that the Nosel and Copnori measure complement each other to provide a reasonable approximation of what the editors should want.

The Nosel measure only weakly enforces the natural order. A very large number of outcomes receive an identical Nosel measure despite the fact that the arrive at different positions in the natural order. From the point of view of the editors, the Nosel measure only penalizes an outcome when the editor has to look at an additional document. It does not take into account, that, for a given value of the position of the last relevant document, the editor may abandon the search for relevant documents before the last relevant document is reached, and may, as a consequence of this action, have a varying number of documents lost in different outcomes that receive the same Nosel penalty.

Such differentiation is provided by the Copnori measure. It strictly enforces the natural order. Each outcomes is assigned a different number and the next worse outcome has a constant additional penalty of one. Moving the last position one further imposes no special additional penalty. But from the point of view of the editors, examining a new document does carry something additional to just one step down in the natural order. From their point of view it has to be a special penalty. Such a extra penalty is provided by the Nosel measure. Therefore it appears best to take a linear combination of the two measures, such as say $\nu\, l(x) + (1 - \nu)\, k(x)$. As long as $0 < \nu \leq 1$, the measure strictly respects the natural order, and gives an extra penalty at each extra document that has to examined in order to find all the relevant documents. We suggest $\nu = 10\%$, but other values are just as acceptable. All they change is the numeric value of the measure. They have no impact on the actual ordering of outcomes.

A Test results

In this table, we report, for a selection of NEP reports, the summary statistics for each measure. a is the Aselt measure, m the Lofop measure, l the Nosel measure, k the Copnori measure, and o the Ponori measure. Reports are ordered by generality. To reduce the size of the table, we omitted three out of four subject reports. For each report and each measure, we see the mean in the line "mean", the minimum in the line "min", the maximum in the line "max", and the standard deviation in the line "dev".

report		a	m	l	k	$o(1.01)$
`nep-mac`	mean	80.21	84.11	21.73	93.19	84.74
	min	67.38	65.58	0.85	39.17	67.29
	max	90.24	93.66	50.55	99.99	94.41
	dev	7.69	8.81	17.43	19.00	9.07
`nep-lab`	mean	81.08	86.16	34.23	98.45	86.96
	min	73.16	73.07	6.96	86.68	71.20
	max	88.88	93.89	79.31	99.99	95.79
	dev	5.70	6.40	20.98	4.16	7.49
`nep-ure`	mean	89.62	93.59	64.09	99.86	94.79
	min	82.63	88.15	36.99	98.71	88.97
	max	97.51	98.72	96.53	99.99	99.25
	dev	5.31	3.70	18.88	0.40	3.50
`nep-eec`	mean	74.72	79.83	26.12	83.03	80.68
	min	55.68	57.14	−1.38	8.50	57.12
	max	84.59	90.94	60.13	99.99	93.03
	dev	9.94	11.30	21.65	30.53	12.08
`nep-tra`	mean	92.52	93.94	65.77	93.28	94.22
	min	75.76	73.46	0.81	34.45	72.68
	max	99.79	99.89	99.24	100.00	99.93
	dev	7.70	8.12	36.01	20.67	8.49
`nep-ino`	mean	88.04	91.31	57.95	91.44	92.40
	min	66.90	66.70	−1.16	14.74	67.77
	max	98.34	99.12	93.36	99.99	99.46
	dev	9.35	9.47	30.28	26.95	9.37
`nep-fmk`	mean	90.45	94.29	71.04	99.96	95.58
	min	79.51	87.71	32.19	99.62	90.48
	max	97.29	98.60	94.71	99.99	99.15

	dev	5.93	3.76	17.39	0.11	3.25
`nep-ifn`	mean	92.14	94.13	57.22	98.07	94.45
	min	83.91	86.67	5.45	81.24	85.69
	max	97.69	98.78	92.02	100.00	99.21
	dev	5.07	4.79	31.32	5.91	5.29
`nep-cba`	mean	87.59	92.49	60.88	99.92	94.05
	min	78.92	86.56	40.33	99.50	88.66
	max	95.28	97.55	93.62	99.99	98.51
	dev	5.34	3.52	17.13	0.15	3.18
`nep-tid`	mean	76.93	84.14	47.17	90.77	86.33
	min	53.25	69.58	−1.56	10.17	71.03
	max	93.39	96.49	87.16	99.99	97.77
	dev	11.42	8.95	24.42	28.32	8.64
`nep-gth`	mean	94.48	96.76	79.96	99.99	97.56
	min	86.33	91.27	49.47	99.96	92.56
	max	99.55	99.76	98.12	100.00	99.85
	dev	3.99	2.62	17.33	0.01	2.34
`nep-cwa`	mean	65.75	68.63	25.00	54.36	70.57
	min	23.64	27.54	−13.11	−81.88	25.32
	max	93.19	96.38	88.88	99.99	97.73
	dev	18.24	21.93	30.73	61.38	20.82
`nep-evo`	mean	89.92	93.97	71.19	99.49	95.35
	min	78.71	85.75	43.05	95.08	87.29
	max	98.10	99.03	97.15	99.99	99.43
	dev	5.94	4.01	16.45	1.54	3.65
`nep-cdm`	mean	69.30	74.01	35.95	80.52	75.88
	min	20.29	−4.61	−14.28	−74.47	−0.08
	max	95.25	97.47	87.38	99.99	98.41
	dev	21.32	29.15	26.40	54.54	28.23
`nep-mfd`	mean	74.58	79.43	32.29	79.67	80.05
	min	57.55	65.30	−4.19	−21.39	67.39

max	92.74	95.96	82.27	99.99	97.24
dev	11.08	10.66	28.94	36.99	11.03

Table A.1: Test results

References

Barrueco Cruz, J. M., Krichel, T., Trinidad Christensen, J. C., 2003. Organizing current awareness in a large digital library, presented at the 2003 Conference on Users in the Electronic Information Environments, in Espoo, Finland and at the II Jornadas de Tratamiento y Recuperación de la Información, in Leganés, Spain, both on September 8, 2003, available at http://openlib.org/home/krichel/papers/espoo.pdf.

Brookes, B. C., 1968. The measure of information retrieval effectiveness proposed by swets. Journal of Documentation 24, 41–54.

Cooper, W. S., 1968. Expected search length: A single measure of retrieval effectiveness based on the weak ordering action of retrieval systems. Journal of American Society of Information Science 19, 30–41.

Cooper, W. S., 1973. On selecting a measure of retrieval effectiveness. part i. the “subjective philosophy” of evaluation. Journal of the American Society for Information Science 24 (87–100).

Ginsparg, P., Houle, P., Joachims, T., Sul, J.-H., 2004. Mapping subsets of scholarly information. Proceedings of the National Academy of Sciences of the USA 101, 5236–5240, available at http://arxiv.org/abs/cs.IR/0312018.

Joachims, T., 1999. Making large-scale svm learning practical. In: Schlkopf, B., Burges, C., Smola, A. J. (Eds.), Advances in Kernel Methods. Support Vector Learning. MIT Press.

Krichel, T., Bakkalbasi, N., 2005. Developing a predictive model of editor selectivity in a current awareness service of a large digital library. Library and Information Science Research 27 (4), 440–452.

Losee, R. M., 1998. Text Retrieval and Filtering: Analytic Models of Performance. Kluwer, Boston.

Shaw, W. M., 1986. On the foundation of evaluation. Journal of the American Society for Information Science 37 (5), 346–348.

Swets, J. A., 1963. Information retrieval system. Science 141 (3577), 245–250.

Van Rijsbergen, C. K., 1974. Foundation of evaluation. Journal of Documentation 30 (4), 365–373.

Vapnik, V. N., 1995. The Nature of Statistical Learning Theory. Springer.