An information-theoretic approach to finding informative noisy tiles in binary databases*  

Kleanthis-Nikolaos Kontonasios†  
Tijl De Bie‡

Abstract
The task of finding informative recurring patterns in data has been central to data mining research since the introduction of the task of frequent itemset mining in [1, 2, 14]. In these seminal papers, the informativeness of a recurring itemset in a binary database was formalized by its support in the database. However, it is now widely recognized that an itemset’s support is not the best measure of its informativeness. Furthermore, recent work has highlighted that the support of an itemset is highly susceptible to noise, such that it may be more appropriate to search for itemsets that recur only approximately. In this paper, we present a new measure of informativeness for noisy itemsets in binary databases within the formalism of tiles [6]. We demonstrate the benefits of our new measure by means of experiments on artificial and real-life data, allowing for objective and subjective evaluation.

1 Introduction
The problem of finding informative sets of items in a binary matrix is a long-standing and important one. It can be decomposed into two tightly interconnected challenges: the adequate formalization of informativeness, and the development of fast algorithms that search a binary database for itemsets that are informative as formalized in this way.

Trading off these two objectives has turned out to be non-trivial. In using the support of an itemset as its informativeness measure, the emphasis has long been on addressing the algorithmic challenge, and it is now well understood how frequent itemsets, defined as itemsets with a sufficiently large support, can be mined efficiently.

However, it has become clear that the support of an itemset is often not adequate as a formalization of informativeness, and alternatives have been proposed. This includes the product of the itemset’s size with its support [6]; an itemset’s ability to compress a database when used as a code-word in a code-table [20]; based on hypothesis testing [21, 7, 5]; and more recently the itemset’s partial-support [16].

In this paper, we adhere to the framework of [6] in considering as patterns of interest the pair of an itemset \( I \) and transaction set \( T \). Such a pair is often referred to as a tile, and we will denote it as \( \tau = (T,I) \). For mathematical convenience, let us represent a database by a binary matrix \( D \in \{0,1\}^{m \times n} \) in this exposition, where \( D(t,i) \in \{0,1\} \) is defined to be equal to 1 iff the transaction \( t \) contains the item \( i \). Then a tile \( \tau = (T,I) \) is said to be present in a database \( D \) if for all pairs of an item \( i \in I \) and a transaction \( t \in T \), the database entry \( D(t,i) = 1 \), i.e. if all transactions from \( T \) carried each item from \( I \). A tile can be approximately present if this is true only for some of the pairs of an item \( i \in I \) and a transaction \( t \in T \). Somewhat abusively, we will refer to the first type of tiles as exact tiles, and the second type of tiles as noisy tiles.

The main result proposed in this paper is a new formalization of informativeness for tiles and noisy tiles, combining ideas from each of the above-mentioned approaches. To do this, we rely on a probabilistic model incorporating prior knowledge about the database as a background model. Then we quantify the Shannon information content of (noisy) tiles with respect to this model. The ratio of this information content over the description length of the (noisy) tile is suggested as a measure of informativeness.

We show how this formalization of informativeness can be generalized toward a set of noisy tiles, referred to as a noisy tiling. Furthermore, we show how a set-covering type of algorithm can be used to search for a tiling that is nearly optimally informative in this sense.

2 Background
Let us first describe various recent attempts to formalize the informativeness of tiles. The newly proposed approach incorporates elements of each of these, and will be discussed in Sec. 4.

---

*This work is supported by the EPSRC grant EP/G056447/1 and the PASCAL2 EU network of excellence. KNK is supported by a Centenary scholarship from the University of Bristol.
†University of Bristol, Intelligent Systems Laboratory.
‡University of Bristol, Intelligent Systems Laboratory.

---

It should be emphasized that all results in this paper could be worded in terms of itemsets and noisy itemsets as well. However, using tiles allows for a more elegant presentation of our results.
2.1 Tiles and tilings

Tiles and tilings were first proposed as a concept in [6]. In the same paper, a measure of informativeness for tiles present in a database was defined, namely as the surface of the tile, i.e., the informativeness measure for a tile $\tau = (T,I)$ was defined as the product $|T| \cdot |I|$ of the number of transactions and the number of items.

For many practical applications, this informativeness measure is much closer to subjective informativeness than the support of an itemset. Indeed, a small itemset that with the same support of a large one is typically less informative than the large one. Multiplying the support by the itemset size resolves this problem.

This measure of informativeness for a tile has elegantly been extended to a measure of informativeness for a tiling defined as a set of tiles, namely as the total surface covered by the tiles. In computing this surface, it is important that parts of the database covered multiple times in the tiling should be counted only once. This ensures that an informative tiling in this sense will typically contain non-overlapping tiles, which are thus minimally redundant.

Regarding tiles as sets of database entries, it can easily be seen that searching for a tiling with a given maximum number of tiles and covering as many database entries as possible is an instance of the budgeted set covering problem. This means that the greedy strategy of iteratively selecting the tile covering as many as possible yet uncovered database entries is approximately optimal.

While yielding a much more useful measure of informativeness and maintaining computational tractability, this approach still has two main disadvantages:

1. Its focus on exact tiles makes it susceptible to noise. Indeed, as pointed out in [22], even small amounts of noise cause large tiles to break up in small pieces.

2. Tiles with individually highly frequent items are considered equally informative as tiles with individually infrequent items, and similarly for transactions. This is despite the fact that tiles over individually infrequent items and small transactions are in a sense more surprising. Individual item frequencies and transaction sizes often belong to the prior knowledge a data miner has about the data, and finding tiles that mainly contain frequent itemsets and large transactions, would amount to rediscovering this. This cannot be the goal.

We will come back to both these issues shortly below.

2.2 Mining noisy frequent itemsets

The first shortcoming of the surface of a tile as informativeness has been addressed in various papers that attempt to mine for noisy frequent itemsets, also known as approximate or fault-tolerant frequent itemsets. While in its more interesting variations this problem is significantly harder than mining noise-free itemsets (or tiles), significant progress has been made in recent years [23, 12, 4, 19, 17]. In particular the method from [17], which allows one to specify the number of tolerated faults in a relative way, was very efficient in our experiments.

The approaches for mining noisy frequent itemsets typically consider the support as informativeness measure. However, conceptually it should be easy to combine these ideas with the ideas from tile mining, considering the surface of a noisy tile as its measure of informativeness, and we will do this for comparison in the experiments in Sec. 5.2. However, the second problem highlighted in Sec. 2.1 remains.

2.3 Swap randomizations

The informativeness of a pattern has often been formalized as its statistical significance under a probabilistic model, as quantified by a p-value. If the probabilistic model substantiates the prior information a data miner has about the data, the significance is indeed likely to be a good measure of the surprise of the data miner when seeing the patterns. This approach has been used successfully in various recent contributions [5, 21, 7, 15, 8].

The contributions in [7, 15, 8] are particularly notable in the context of this paper. They rely on the idea of randomizing databases while respecting constraints that reflect the prior knowledge. An important type of prior knowledge considered in these papers are on the value of the item and transaction marginals: the number of times each item is supported by a transaction, and the size of each transaction. They show that databases can be randomized while keeping this information constant by using a chain of elementary randomization operations called swaps.

Using the swap randomization approach, it is possible to perform hypothesis tests and compute the statistical significance of any well-defined pattern in a database. While the details would need to be worked out, this has the potential to resolve most of the issues highlighted in Sec. 2.1, so as to come up with a suitable measure of informativeness for noisy tiles.

However, the swap randomization approach, being a computation intensive technique, has other disadvantages. In particular, it cannot be used to analytically define measures of informativeness, which makes it harder to directly mine for informative noisy tiles.

2.4 KRIMP

In this paper, we will propose a new approach sharing considerable benefits with the swap
randomization approach, while being entirely analytical. Based on this, we will propose a new measure of informativeness. Our measure is inspired by information theoretic ideas and the Minimum Description Length (MDL) principle. Therefore, we should highlight one other interesting approach to defining informativeness measures for itemsets: the method KRIMP [20].

KRIMP searches for itemsets that allow one to compress the database as much as possible. It does this by constructing a code-table consisting of the selected itemsets, and using this code table to encode the database. The more concise the code table and the better it allows one to reconstruct the database, the shorter the overall description length, and the better the code table is judged to be. Pioneered in KRIMP for the context of itemset mining, the MDL idea has since been used successfully for various other tasks, such as for event detection [13] and for the identification of relational patterns [11].

However, as opposed to the swap randomization approach to finding informative itemsets, KRIMP is not designed to take prior information into account. We believe this is critical for many applications, in particular if the patterns found are meant for human consumption and in some cases also if they are meant for further processing. Another disadvantage is algorithmic: searching for the best code table is a hard problem, and the use of heuristics is inescapable.

3 The maximum entropy model for binary databases

As pointed out in Sec. 2.3, incorporating prior knowledge into a probabilistic background model for the data is a promising approach for the statistical assessment of patterns found in data, and hence for the formalization of meaningful measures of informativeness. In the case of binary databases, such prior information often takes the form of constraints on the item and transaction marginals. Why this is meaningful in practice has successfully been argued in [7].

Swap randomizations allow one to sample approximately uniformly from the set of all databases with fixed item and transaction marginal. However, specifying this distribution only implicitly, they do not allow one to analytically specify or compute any statistical measure of significance or informativeness of any given pattern (such as a noisy tile) found in the database.

In this paper, we follow the approach introduced in [3] to fit an explicit probabilistic model known as the MaxEnt model to the database, respecting the same kinds of constraints, albeit in expectation. Let us first briefly summarize how to find this distribution. In Sec. 4 we will then show how it can be used to define a new measure of informativeness for noisy tiles.

3.1 Prior information as constraints on the probability distribution Let us denote the probabilistic model for the database $D \in \{0,1\}^{m \times n}$ as $P$. Then, the constraints on the expected transaction and item marginals can be written as:

$$\sum_{D \in \{0,1\}^{m \times n}} P(D) \cdot \left( \sum_{i} D(t,i) \right) = mp_t,$$

$$\sum_{D \in \{0,1\}^{m \times n}} P(D) \cdot \left( \sum_{i} D(t,i) \right) = np_t.$$

Here, $mp_t$ is used to denote the required expected marginal of item $i$ (i.e. $p_i$ is the expected proportion of transactions that contain item $i$), and $np_t$ denotes the expected marginal of transaction $t$ (i.e. $p_t$ is the expected proportion of items contained in transaction $t$).

Besides these constraints, a probability distribution also needs to be positive and normalized:

$$P(D) \geq 0, \forall D \in \{0,1\}^{m \times n},$$

$$\sum_{D \in \{0,1\}^{m \times n}} P(D) = 1.$$

For non-trivial cases, these constraints do not uniquely specify the probability distribution $P$. So the question remains which of the feasible solutions should be chosen.

3.2 The Maximum Entropy distribution In [3] it was argued that the Maximum Entropy (MaxEnt) distribution among those satisfying the constraints from Eqs. (3.1-3.4) satisfies some desirable properties. Exhibiting the largest amount of uncertainty, it incorporates no additional biases besides the prior information (see also [9] for a motivation of this rationale). Furthermore, the MaxEnt distribution is robust in the sense that if a Shannon-optimal code is designed based on it, the expected code length for randomly drawn data from the (unknown) true distribution is minimal in the worst case over all true distributions that satisfy the same prior information constraints.

The entropy $E(P)$ of the distribution $P$ is given by:

$$E(P) = - \sum_{D \in \{0,1\}^{m \times n}} P(D) \log P(D).$$

As shown in [3], this objective can be maximized efficiently subject to the constraints from Eqs. (3.1-3.4), and we will further refer to this distribution as the MaxEnt distribution. Without giving details of
the derivation, let it suffice to state that this MaxEnt distribution is of the form:

\[
P(D) = \prod_{t,i} \frac{\exp(D(t,i)(-\lambda_t - \mu_i))}{1 + \exp(-\lambda_t - \mu_i)} ,
\]

where \(\lambda_t\) and \(\lambda_i\) can be found efficiently as shown in [3].

3.3 Discussion of the MaxEnt distribution

From the shape of the MaxEnt distribution in Eq. (3.6), it can be seen that it is a product distribution of independent Bernoulli random variables, one for each database entry \((t,i)\). The success probability at entry \((t,i)\) is given by \(\frac{\exp(-\lambda_t - \mu_i)}{1 + \exp(-\lambda_t - \mu_i)}\).

Interestingly, the MaxEnt distribution is invariant under swap randomizations. More exactly, the probability for a database \(D\) under the MaxEnt model is the same as for any database that can be achieved by swap randomizing \(D\). This means that the MaxEnt model is in some sense a close but analytically expressed surrogate for the uniform distribution over all databases with given item and transaction marginals. The proof of this simple but interesting fact is given in [3].

The MaxEnt distribution for binary databases as used here is formally identical to the Rasch model [18], used in psychometrics. There, the transactions corresponds to subjects in a psychological test, and the items to questions in a questionnaire to which the subjects are subjected. Then, \(D(t,i)\) is equal to 1 if the subject \(t\) answered the question \(i\) correctly. The negative of the variable \(\lambda_t\) then corresponds to the ability of subject \(t\), and the variable \(\mu_i\) represents the difficulty of question \(i\). However, the Rasch model is used for totally different purposes as those in this paper. Interestingly, it is also motivated in a different way, not relying on the MaxEnt principle.

4 Formalizing informativeness

Given a probabilistic model for data, the Shannon information content can be computed. This is the number of bits (using logarithms base 2) required to encode the data when an optimal Shannon code is used, and is equal to the negative log probability for the data.

As we will show below, it is also possible to quantify the Shannon information content in localized patterns in the database, such as tiles or noisy tiles. Since the MaxEnt model incorporates prior knowledge of the data miner, tiles or noisy tiles with a higher information content with respect to this model would be more informative to a data miner with this prior information.

Of course, (noisy) tiles themselves need to be described, and it is no use to transmit a lot of information if it is described in an inefficient way. Therefore, below we will not only focus on the information content of tiles, but also on the length of their description. A trade-off between these two quantities is the informativeness measure we propose in this paper.

Let us formalize and further motivate these ideas first for tiles, and subsequently for the more complex case of noisy tiles.

4.1 Tiles Let us first introduce a new informativeness measure for tiles. Then we will generalize it to tilings (collections of tiles). Then we will demonstrate how informative tilings can be found algorithmically.

Informativeness of a tile Given the MaxEnt model, the Shannon information content in \(D(t,i)\) being equal to 1 is equal to \(-\log\left(\frac{\exp(-\lambda_t - \mu_i)}{1 + \exp(-\lambda_t - \mu_i)}\right)\); minus the success probability of the corresponding Bernoulli random variable. Therefore, and thanks to the independence of the database entries under the MaxEnt model, the total information content in the statement that a tile \(\tau = (T,I)\) is present in \(D\) is equal to:

\[
\text{InformationContent}(\tau) = - \sum_{t \in T, i \in I} \log \left(\frac{\exp(-\lambda_t - \mu_i)}{1 + \exp(-\lambda_t - \mu_i)}\right).
\]

This is the number of bits that would be required to transmit this information using an optimal Shannon code designed for the MaxEnt model. Interestingly, the information content of a tile can be regarded a weighted version of the surface of a tile.

In order to quantify the description length of a tile, we need to agree on an encoding scheme for tiles, i.e. for the itemset \(I\) and the transaction set \(T\) defining the tile. We suggest to encode the itemset of a tile using an optimal Shannon code with respect to a probabilistic model for sets of items, in which item presences are independent of one another. In this model we set the probability of item \(i\) appearing equal to \(p_i\), the proportion of transactions containing that item. Similarly, we assume a model of independence for transaction sets, where the probability of a transaction is \(p_t\). Then, any itemset \(I\) can be encoded by using a number of bits equal to \(-\sum_{i \in I} \log(p_i) - \sum_{i \in I} \log(1 - p_i)\).

Similarly, a transaction set \(T\) can be encoded by using \(-\sum_{t \in T} \log(p_t) - \sum_{t \in T} \log(1 - p_t)\) bits. The total description length of a tile is therefore:

\[
\text{DescriptionLength}(\tau) = - \sum_{i \in I} \log(p_i) - \sum_{i \in I} \log(1 - p_i) - \sum_{t \in T} \log(p_t) - \sum_{t \in T} \log(1 - p_t).
\]
Note that typically databases are sparse, such that
$p_i$ and $p_i$ are small and the terms $- \sum_{i \in I} \log (1 - p_i)$
and $- \sum_{i \in T} \log (1 - p_i)$ do not contribute much to
the description length. Therefore, to a good approximation,
the two components in the description length of a tile
are proportional to the size of the itemset and the size
of the transaction set respectively.

From the perspective of a data miner, we argue that
a tile would be regarded as more valuable if the information
it conveys is maximally compressed. Therefore,
we suggest as a new measure of informativeness for tiles
the ratio of the information content over the description
length, and we call it the information ratio:

\[
\text{InformationRatio(\tau)} = \frac{\text{InformationContent(\tau)}}{\text{DescriptionLength(\tau)}}.
\]

This measure quantifies the density of information in a
tile.

**Informativeness of a tiling** The information ratio as a measure of informativeness of a tile can easily
be generalized to quantify the informativeness of a tiling
\( \tau = \{ \tau_1, \tau_2, \ldots, \tau_K \} \), a collection of tiles. To this end,
the information content and the description length need
to be generalized toward equivalents for tilings. The information content is generalized as follows:

\[
\text{InformationContent(\tau)} = \sum_{t,i : \exists \tau = (T,I) \in \tau, t \in T, i \in I} \log \left( \frac{\exp(-\lambda_t - \mu_i)}{1 + \exp(-\lambda_t - \mu_i)} \right),
\]

the sum of the Shannon information in each entry
covered by any of the tiles from \( \tau \). Note that entries
covered by two different tiles in a tiling should be
counted only once.

The description length on the other hand is
generalized more directly, as:

\[
\text{DescriptionLength(\tau)} = \sum_k \text{DescriptionLength}(\tau_k).
\]

The information ratio of a tiling can be generalized
as the ratio of the information content over the description
length of the tiling.

**Searching for an informative tiling, and the
weighted budgeted maximum covering problem**
A practical formalization of the data mining task would
be as follows. Given a finite processing capability of the
data miner (the length of the message the data miner receives),
convey as much interpretable information as possible about the data. In the present framework,
this would amount to searching for the tiling that has the
largest information content subject to an upper bound
on its description length.

Given a possibly large number of tiles from which
the tiling is selected, this problem is an instance of
the weighted budgeted maximum coverage problem
(WBMSC), a variant of the well-known set covering
problem. Here, a number of sets of elements is given
with a cost associated to each set and a weight to each element.
The task is to select a collection of these sets
such that the sum of their costs is smaller than a given
budget, and the sum of the weights of elements in the
union of the selected sets is as large as possible. Applied
to our problem, the elements are the database entries,
and the sets are the sets of entries covered by each tile.
The cost of a tile is its description length, while the
weight of each entry is its Shannon information content.

The WBMSC problem can be approximated well by
means of a greedy algorithm, with a guaranteed
approximation ratio of $1 - \frac{1}{e}$ [10]. Note that a similar strategy
was used in [6], where the problem of finding the tiling
covering the largest number of database entries was
observed to be an instance of the (unweighted) budgeted
maximum coverage problem.

In general, the greedy algorithm for the WBMSC
iteratively selects the set for which the increase in the
total weight of the covering divided by the cost of the
set is maximal. Applied to the tile-mining setting in
this paper, this means selecting the tile for which the
increase in information content of the tiling divided
by the description length of the tile is maximal. In
other words, it iteratively selects the tile that condenses
information most succinctly. Formally, this means that
each subsequent tile is selected based on what we will
call the conditional information ratio:

\[
\text{InformationRatio}^+(\tau) = \frac{\text{InformationContent}^+(\tau)}{\text{DescriptionLength}(\tau)}.
\]

Here,

\[
\text{InformationContent}^+(\tau) = \sum_{t,i : \exists \tau = (T,I) \in \tau, t \in T, i \in I \text{ uncovered}} \log \left( \frac{\exp(-\lambda_t - \mu_i)}{1 + \exp(-\lambda_t - \mu_i)} \right),
\]

where the sum is over all entries covered by the tile that
are not yet covered by tiles already added to the tiling
in previous iterations of the greedy algorithm.

Note that the greedy algorithm can be stopped after
any desired number of iterations. Hence, one could stop
selecting additional tiles as soon as a desired maximum
description length is reached. However, rather than
pre-specifying this maximum, one could use the greedy
algorithm to simply rerank all candidate tiles and decide
later at which level to cut off. The approximation
guarantee still holds no matter which cut-off is chosen.

Hence, the scheme of the algorithm is as follows:
1. Run any tile mining algorithm, or any algorithm for mining itemsets that are then complemented by their supporting transactions to form tiles. This is the set of candidate tiles.

2. Iteratively select the tile with the largest conditional information ratio from the candidate tiles, and append it to a ranked list.

3. For any given cut-off corresponding to a threshold on the total description length, the top tiles in the ranked list approximately maximize the information content given that threshold, with a guaranteed approximation ratio of $1 - \frac{1}{e}$.

4.2 Noisy tiles

Noisy tiles are tiles where part of the database entries covered by the tile may not be equal to 1. This affects both the information content and the description length. Other than these modifications, the above discussion for noiseless tiles remains applicable directly to noisy tiles. So let us focus only on the generalization of the information content and description length toward noisy tiles.

**Information content of a noisy tile**

This is computed by summing the Shannon information content of each of the database entries covered by the tile:

$$\text{InformationContent}(\tau) = - \sum_{t \in T, i \in I : D(t, i) = 1} \log \left( \frac{\exp(-\lambda_t - \mu_i)}{1 + \exp(-\lambda_t - \mu_i)} \right) - \sum_{t \in T, i \in I : D(t, i) = 0} \log \left( \frac{1}{1 + \exp(-\lambda_t - \mu_i)} \right).$$

In sparse databases, the terms in the second sum are much larger than those in the first sum, such that the information content decreases the more database entries $D(t, i)$ covered by the noisy tile are equal to 0.

**Description length of a noisy tile**

To describe a noisy tile $\tau = (T, I)$, not only do its itemset $I$ and its transaction set $T$ need to be described. Also the number of zeros it covers in the database, as well as their positions, need to be described. The former part of the code is identical to the description for noiseless tiles, but the rest is new.

The description of how many zeros there are will increase the description length by at most the logarithm of the surface of the tile: $\log(|T| \cdot |I|)$. This is because the number of zeros is necessarily between 0 and $|T| \cdot |I|$. Given this number of zeros $n_0$, the number of tile constellations is upper bounded by $\binom{|T| \cdot |I|}{n_0}$, such that a number of bits equal to $\log \left( \binom{|T| \cdot |I|}{n_0} \right)$ is required to describe it. Using Stirling's approximation for the factorial, this can be shown to be equal to $|T| \cdot |I| \cdot H \left( \frac{n_0}{|T| \cdot |I|} \right) + o(|T| \cdot |I|)$, where $H(p) \triangleq -p \log(p) - (1-p) \log(1-p)$ stands for the entropy of a Bernoulli random variable with success probability $p$. Note that this is typically much larger than the description length $\log(|T| \cdot |I|)$ required for the number of zeros, so we can approximate the description length of a noisy tile as:

$$\text{DescriptionLength}(\tau) = - \sum_{t \in T} \log (p_t) - \sum_{i \in I} \log (1 - p_i) - \sum_{t \in T} \log (1 - p_t) + |T| \cdot |I| \cdot H \left( \frac{n_0}{|T| \cdot |I|} \right).$$

Clearly, the closer the fraction of zeros $\frac{n_0}{|T| \cdot |I|}$ in a tile is to $\frac{1}{2}$, the larger the entropy term and the larger the description length of the noisy tile.

The **information ratio and noisy tilings**

The information ratio introduced above for noiseless tiles can be generalized directly for noisy tiles. Similarly, the concept of a tiling can be generalized toward a noisy tiling, as well as its associated description length and information content. Also the WBMSC approach to find an optimal tiling can be used for noisy tilings.

5 Experiments

A series of experiments have been developed and performed in order to evaluate the performance of the proposed informativeness measure. These experiments are divided, by the nature of the datasets used, in artificial and real-life experiments. The first step in our algorithm involves the running of a noisy tile/itemset mining algorithm.

The artificial experiments provide a chance to examine in a controlled environment many aspects of the behavior of the proposed measures, making in this way the interpretation of an informative itemset proposed in this paper as clear as possible.

On the other hand, applying our method to real-life datasets provides the opportunity for subjective evaluation of the proposed measure of informativeness. Since it is hard to subjectively evaluate the informativeness of noisy tiles on widely used datasets such as the retail, mushroom, or chess datasets, we have opted to make use of a text corpora instead. The mapping from texts to binary databases is done by regarding each text as a transaction and each word as an item.

We will conclude the experimental evaluation with some remarks about computational efficiency.
Figure 1: Information Ratio(A), Information Content(B), Description Length(C) for different levels of noise. A $10 \times 5$ tile is embedded in a $1000 \times 100$ database generated with $p = 0.05$.

Figure 2: Hypothesis Testing Experiments – Empirical p-values for different levels of noise. A $10 \times 5$ tile is embedded in a $1000 \times 100$ database generated with $p = 0.05$.

5.1 Artificial data The experiments on artificial data are meant to study whether tiles artificially embedded in an otherwise random binary database would be retrieved by our method. Furthermore, they allow us to study the noise resilience of the proposed measure of informativeness. We have organized the experiments around three questions we want to answer about the information ratio as a measure of informativeness.

**Question 1: Does an artificially embedded tile have a high information ratio?** In order to address this question artificial binary databases of size $n \times m$ were created, and we report results for $n = 1000$ and $m = 100$. Each entry in the database was a result of an independent Bernoulli trial with probability that a specific item is included in a specific transaction chosen equal to $p = 0.05$ for all $n \cdot m$ database entries. Subsequently, an exact tile was embedded manually in the database. After calculating the InformationRatio, InformationContent and DescriptionLength for the initial tile, a tile’s element chosen at random is switched to zero and the calculation is repeated. Iterations of this scheme, with a zero always replacing a one, produce the values of the quantities of interest for different noise levels in the tile.

The results show an almost linear decrease of the InformationRatio value with the increase of noise in the tile (see Fig. 1(A)). This is a desired property for our interestingness measure since it is intuitively satisfying that between an exact tile and its noisy versions, the exact tile is the most interesting one.

Calculating the DescriptionLength of a tile using an entropy function defines no preference in encoding ones over zeros, in that the symmetry of the entropy function assigns the same DescriptionLengths to complementary tiles (see Fig. 1(C)). At first sight, it seems like this may result in undesired behavior for the InformationRatio measure. However, as experiments in Fig. 1(A) show, this is not the case due to a simultaneous and more drastic decrease of InformationContent of the tile (see Fig. 1(B)).

**Question 2: Is the InformationRatio of an embedded tile statistically significant?** After calculating the InformationRatio of a tile, it is natural to consider whether the calculated value deviates significantly from the one calculated for a tile in the same position but in a binary database created at random according to the MaxEnt model. Only if the answer to this question is positive is it meaningful to use the InformationRatio to distinguish informative from non-informative noisy tiles. Doing this for a range of noise levels in the artificially embedded noisy tile will allow us to understand how much noise can be tolerated for the InformationRatio to remain effective.

To address this question 1000 databases were sampled from the MaxEnt distribution fitted to the database in which a noisy tile is embedded (this original database was generated as explained under Question 1 with the same values for $p$, $n$, and $m$). Then, the InformationRatio values for the tile were calculated for each of these 1000 databases. The statistical significance was
measured by computing the empirical p-value, defined as the fraction of the 1000 databases for which the InformationRatio was larger than for the original database in which the tile is embedded. This experiment was repeated for different noise levels in the embedded noisy tile, in order to compute the empirical p-value over the entire range of noise levels.

The results, plotted in Fig. 2, show that the InformationRatio values are statistically highly significant even for very high noise levels.

**Question 3: Can an embedded tile be recovered?** So far we have not attempted to discover the embedded tile, all we have argued for is that the InformationRatio is suitable for distinguishing an embedded noisy tile from the background. Here, we again embed a noisy tile in the database. However, this time the approximate frequent itemset mining algorithm from [19] is applied in order to rediscover the tile along with all other existing tiles in the database. Subsequently our method is applied in order to create a sorted list of the discovered tiles according to the InformationRatio values. The question is whether the embedded tile is recovered by our method, i.e. ranks highly in the sorted list.

Our experiments showed that the embedded tile was recovered in most of the cases tried. The different experimental settings included different values of the Bernoulli parameter \( p \) for the database generation, and different sizes of the embedded tile and parameter settings in the approximate frequent itemset mining algorithm. Let us give a brief overview of some illustrative results.

Fig. 3 and Fig. 4 present some interesting properties of the proposed measure on databases as generated under Questions 1 and 2. In Fig. 3 the tiles produced by the frequent itemset mining algorithm were only the embedded tile and its submatrices. The method succeeds not only in ranking the embedded tile first but also in making the InformationRatio ranking a refinement of the ranking according to the surface of the tile. It is clearly seen in Fig. 3 that every major decrease in the InformationRatio value corresponds to a decrease in the cardinality of the corresponding tile, while at the same time the InformationRatio values for itemsets of equal cardinality continue to vary slightly.

In the case of Fig. 3 ranking according to the cardinality of the tile would have lead to approximately the same results as ranking based on the InformationRatio. The experiment in Fig. 4 presents a different case. Here the parameter settings for the approximate frequent mining algorithm are loosened and the binary database created was less sparse than the one used for Fig. 3 (\( p = 0.1 \) instead of \( p = 0.05 \)). As a consequence the tiles produced are not only submatrices of the initially embedded tile but also combinations of areas of the initial tile with other columns of the binary matrix. In this case, simply ranking the tiles by cardinal-
ity of the tile would have resulted in introducing tiles produced by chance to the top places of the ranking. However the results presented in Fig. 4 show that uninteresting tiles can be placed low in the ranking based on their InformationRatio even if they have a large cardinality. This suggests that the InformationRatio is more robust than a method that is purely based on the surface (cardinality) of tiles. A more subjective comparison with the tile-based approach is given in Sec. 5.2.

5.2 Real-life data Textual datasets are easily accessible for subjective evaluation of the proposed measure of informativeness. In particular we report the results in three datasets taken from papers' abstracts from ICDM up to 2007 and a data mining related selection of Pubmed papers as detailed in [3], as well as all KDD abstracts between 2001 and 2008. Stop word removal and stemming was applied. Every abstract is considered a transaction in a binary database, and every stemmed word an item. The characteristics of the datasets are presented in Tables 1. Here we used the approximate frequent itemset mining algorithm from [17], and the parameter settings used are shown in Tab. 2.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#items</th>
<th>#transactions</th>
<th>#noisy tiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pubmed</td>
<td>12,661</td>
<td>1,683</td>
<td>13,153</td>
</tr>
<tr>
<td>ICDM</td>
<td>4,974</td>
<td>858</td>
<td>26,155</td>
</tr>
<tr>
<td>KDD</td>
<td>6,154</td>
<td>843</td>
<td>208,532</td>
</tr>
</tbody>
</table>

Results for the InformationRatio Fig. 5 plots the InformationRatio for the 30 tiles with the highest InformationRatio. The word sets for the noisy tiles with the 10 highest InformationRatio are presented in the leftmost column of Tables 3 for ICDM, Pubmed, and KDD datasets. These lists do contain intuitively informative itemsets, but they are highly redundant. In order to remove the redundancy the WBMSC algorithm is applied. The rightmost column of the same Table shows the word sets for the 10 noisy tiles that are top-ranked by the WBMSC algorithm, and the redundancy is decreased considerably. Most of the word sets for noisy tiles in the tiling present major topics and concepts of data mining (ICDM, KDD) and data mining applied to medical and biological problems (Pubmed).

Empirical comparison Comparing informativeness measures with other approaches is difficult for two reasons. First, the most sensible way to assess whether an informativeness measure is more useful than another one is by subjective comparison. And second, to our knowledge no other informativeness measure other than the support has been defined for noisy itemsets/tiles. However, sorting noisy itemsets purely based on their support will almost trivially lead to bad results, with the top-ranked itemsets being very small. Furthermore, due to the lack of a redundancy-reducing mechanism the top-ranked results are likely to be highly redundant.

For a fairer comparison, we therefore consider the surface of a noisy tile as its informativeness measure, as a trivial adaptation of the informativeness measure from [6] (originally developed for noise-free tiles). The top-ranked noisy tiles in this measure are shown in Tab. 4. Clearly, the top-ranked tiles contain words that are individually highly frequent, and carry little information about topics in the corpora investigated. The reason for this is that the prior information about the individual word frequencies is not taken into account in the tile surface as an informativeness measure, as we forewarned in Sec. 2.2.

5.3 Computational study The computational bottleneck in all our experiments was invariably the noisy itemset mining, while the fitting of the MaxEnt model. The computation of the InformationRatio for all candidate tiles, and the application of the WBMSC method were considerably faster. For example, computationally the most challenging experiment reported in this paper is the KDD experiment. Here, on a standard laptop with 2Gb RAM and a 2GHz dual core Pentium, the noisy itemset miner took approximately 15 minutes, whereas the MaxEnt model fitting took approximately 4 minutes. Given that the computation time of the proposed method was always significantly less than the time required by the noisy itemset mining step, we judged that including further details on computation times was not sufficiently interesting to be warranted within the available space in this conference paper.

6 Conclusions
Finding informative patterns in data is central to data mining. An often studied case is the problem of finding informative sets of items in a binary database.

Much progress has been made in recent years toward designing efficient methods that search for itemsets that make intuitive sense, by formalizing the notion of informativeness of an itemset in various ways. We have highlighted some key approaches and pointed out their most important strengths and weaknesses.

In this paper, we adhered to the tile formalism to think about itemsets and transaction sets in binary databases. We then proposed a new approach to quantifying the informativeness of tiles and noisy tiles. Our approach exploits the ability to fit explicit models to bi-
Table 2: Parameters setting for the noisy frequent itemset miner [19].

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Row error tolerance</th>
<th>Column error tolerance</th>
<th>Pattern error tolerance</th>
<th>Support threshold</th>
<th>$X_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pubmed</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>5% #transactions</td>
<td>10</td>
</tr>
<tr>
<td>ICDM</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>5% #transactions</td>
<td>10</td>
</tr>
<tr>
<td>KDD</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>5% #transactions</td>
<td>10</td>
</tr>
</tbody>
</table>

![Image](image_url)  
Figure 5: InformationRatio rankings for (a) Pubmed, (b) ICDM and (c) KDD.

nary databases, incorporating relevant prior knowledge [3]. The informativeness measure used then contrasts the found (noisy) tiles with this probabilistic model using information theoretic quantities.

Finding informative (noisy) tiles defined in this novel way reduces to an instantiation of the weighted budgeted maximum set coverage problem. Finding a guaranteed near-optimal solution to this problem can be done extremely efficiently using a greedy algorithm.

We believe this is only one possible application of the MaxEnt model introduced in [3], and we see this paper as the first step in this direction. However, it is likely that the MaxEnt model can be used to define other measures of informativeness, with different properties, for example based on ideas from hypothesis testing or by model fitting. Furthermore, it could be used to formalize the informativeness of other patterns in binary databases, such as association rules, as well as for patterns in other types of data for which the MaxEnt model can be found. This is the subject of future work.

Acknowledgements The authors are indebted to A.K. Poernomo and R. Gupta for their kind help with using their tools from [17, 19], and to Bart Goethals for interesting discussions about prior versions of this work.

References

Table 3: Top-10 informative itemsets ranked according to the InformationRatio (left column) and the conditional InformationRatio (right column) for ICDM (top), Pubmed (middle) and KDD (bottom).

<table>
<thead>
<tr>
<th>Top-10 items ranked using InformationRatio</th>
<th>Top-10 items ranked using cond. InformationRatio</th>
</tr>
</thead>
<tbody>
<tr>
<td>vector machine support</td>
<td>vector machine support</td>
</tr>
<tr>
<td>discover frequent efficient pattern mine algorithm</td>
<td>discover frequent efficient pattern mine algorithm</td>
</tr>
<tr>
<td>association rules database mine algorithm</td>
<td>association rules database mine algorithm</td>
</tr>
<tr>
<td>frequent database pattern large mine algorithm</td>
<td>frequent pattern emerging set</td>
</tr>
<tr>
<td>association rules database mine algorithm</td>
<td>frequent itemset</td>
</tr>
<tr>
<td>frequent database efficient pattern mine algorithm</td>
<td>synthetic real</td>
</tr>
<tr>
<td>frequent study problem mine algorithm</td>
<td>dimensional high cluster model</td>
</tr>
<tr>
<td>frequent database pattern large mine</td>
<td>serl time</td>
</tr>
<tr>
<td>frequent study efficient pattern mine algorithm</td>
<td>decision tree classifier</td>
</tr>
<tr>
<td>neural networks</td>
<td>neural networks</td>
</tr>
<tr>
<td>neural networks</td>
<td>microarray gene expression</td>
</tr>
<tr>
<td>microarray express gene method analysis result mine</td>
<td>play role mine data</td>
</tr>
<tr>
<td>microarray express gene method analysis result mine</td>
<td>throughput high mine</td>
</tr>
<tr>
<td>microarray express gene method analysis result mine data</td>
<td>this-website available</td>
</tr>
<tr>
<td>play role mine data</td>
<td>knowledge discovery</td>
</tr>
<tr>
<td>microarray express tool gene analysis mine</td>
<td>clinical patient</td>
</tr>
<tr>
<td>microarray express tool gene analysis data</td>
<td>machine learn mine data</td>
</tr>
<tr>
<td>microarray express approach gene analysis result data</td>
<td>conclusion perform study method analysis result data</td>
</tr>
<tr>
<td>frequent pattern efficient mine algorithm</td>
<td>frequent pattern efficient mine algorithm</td>
</tr>
<tr>
<td>rule association</td>
<td>rule association</td>
</tr>
<tr>
<td>frequent pattern set study mine</td>
<td>frequent pattern efficient mine paper data</td>
</tr>
<tr>
<td>frequent pattern efficient mine paper data</td>
<td>frequent pattern efficient mine propos</td>
</tr>
<tr>
<td>vector support classification paper data</td>
<td>frequent pattern efficient mine show</td>
</tr>
<tr>
<td>frequent pattern effective mine propos</td>
<td>vector support classification algorithm data</td>
</tr>
<tr>
<td>vector support</td>
<td>vector support</td>
</tr>
<tr>
<td>frequent pattern efficient mine paper</td>
<td>frequent pattern efficient mine paper</td>
</tr>
</tbody>
</table>

International Conference on Knowledge Discovery and Data Mining (KDD09), 2009.


[19] R.Gupta, G.Fang, B.Field, M. Steinbach, and V. Ku-
Table 4: Top-10 informative itemsets ranked according to the surface of their corresponding tile (left column) for ICDM(top), Pubmed(middle) and KDD (bottom).

<table>
<thead>
<tr>
<th>Top-10 itemsets ranked according to the surface of their corresponding tile.</th>
</tr>
</thead>
<tbody>
<tr>
<td>set mine base algorithm</td>
</tr>
<tr>
<td>problem result base algorithm</td>
</tr>
<tr>
<td>approach result base algorithm</td>
</tr>
<tr>
<td>problem approach base algorithm</td>
</tr>
<tr>
<td>approach mine base algorithm</td>
</tr>
<tr>
<td>problem mine result algorithm</td>
</tr>
<tr>
<td>problem mine base algorithm</td>
</tr>
<tr>
<td>approach mine result algorithm</td>
</tr>
<tr>
<td>mine result base algorithm</td>
</tr>
<tr>
<td>problem set base algorithm</td>
</tr>
</tbody>
</table>

| method result mine data  |
| method analysis result mine data  |
| analysis result mine data  |
| base method result mine data  |
| base analysis result data  |
| base analysis result mine data  |
| mine data  |
| base method analysis mine data  |
| study analysis result mine data  |
| base result mine data  |

| base propos algorithm paper data  |
| method propos algorithm paper data  |
| result base propos paper  |
| method base propos paper data  |
| method base algorithm paper data  |
| result propos algorithm paper data  |
| result base algorithm paper data  |
| problem propos algorithm paper data  |
| show propos algorithm paper data  |
| set propos algorithm paper data  |

---


