# A BINARY-CATEGORIZATION APPROACH FOR CLASSIFYING MULTIPLE-RECORD WEB DOCUMENTS USING A PROBABILISTIC RETRIEVAL MODEL

by

Quan Wang

A thesis submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computer Science

Brigham Young University

November 2001

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

The World Wide Web contains a tremendous amount of information. It is challenging to (i) determine which Web documents are relevant to a user query and (ii) rank them according to their degree of relevance to the user query. Many efforts have been made to study different information retrieval (IR) approaches including the Boolean model and the Vector Space model (VSM) [BYRN99], to solve these problems. The Boolean model is based on binary decisions without a grading scale and hence cannot be used for relevance ranking. VSM overcomes the limitation of the Boolean model by (i) assigning predefined non-binary weights to index terms in documents, as well as in user queries, and (ii) computing the degree of similarity between a document and a user query using the assigned weights. Since there is an intrinsic limitation that these two models cannot handle any dependent relationships among index terms, they suffer from the problem of oversimplification. Furthermore, (i) retrieved evidences (e.g., keywords or index terms) are assumed to be mutually independent, and (ii) no semantics (e.g., closely-related index terms) of documents to be examined or in user queries to be processed is considered.

Probabilistic models, on the other hand, retrieve information by applying probability theory. Probabilistic models are based on the *probability ranking principle* [Rob77] where retrieved documents are ranked according to their relevance probabilities with respect to a user query [CLvR98]. Early probabilistic models, such as the *binary independence indexing model* (BII) and the *binary independence retrieval model* (BIR) use mutual independence assumptions exclusively to make the models computationally workable, which by no means provides any justification to do so and inevitably causes distortion [CLvRC98]. Another type of probabilistic models uses the statistical techniques, such as the regression analysis [CLvRC98]. These approaches use none or weaker independence assumptions than BII and BIR, and are referred to as the "model-free" approach [CLvRC98]. One of the "model-free" approaches is the Darmstadt indexing approach (DIA) proposed by Fuhr [CLvRC98], where relevance probability estimates are calculated by applying standard polynomial regression (i.e., use a polynomial function to fit probability distribution) on a learning process (i.e., training process) over a sample space (i.e., training document set). The experimental results show that DIA is superior to the other index-term-based retrieval approaches [CLvRC98]. However, as indicated by Cooper [CGD92], it is not appropriate to use polynomial regression analysis in probability estimates since the values of a dependent variable in the polynomial regression analysis change gradually but the relevance probability values are discrete[1]. A more appropriate approach, initially proposed by Cooper, is to replace polynomial regression by logistic regression in relevance probability estimates [CGD92]. Furthermore, we can enhance relevance probability estimate by analyzing correlation relations among index terms using logistic regression. In this thesis, we adopt a probabilistic model that follows Cooper's direction.

Our probabilistic model also tries to weaken the independence assumptions as well and is capable of adding correlation relations in relevance probability calculation. When we perform logistic regression analysis on an index term, we also obtain the statistical information, i.e., the $p$-value, associated with it. By definition, the $p$-value is the probability that a test is more extreme than the test could have occurred by chance alone [NWH83]. In our probabilistic retrieval model, which is based on logistic regression, the $p$-value indicates (i) how significance an index term is in the determination of the relevance probability of a document with respect to a user query, and (ii) how confident we are in these statistical results. We use the $p$-values of index terms in a document to determine which index terms are significant and should be included in computing the relevance probability of the document with respect to a user query. This process is critical because adding statistically insignificant index

[1]Documents in a given set are divided into two groups. The documents in one group are relevant to a query and each one in the group has the relevance probability of 1, whereas documents in the other group are irrelevant and are assigned the probability of 0.

terms in relevance probability calculation can potentially cause over-fitting and as a result the performance of our probabilistic model deteriorates.

Oversimplification, such as the absolute independence assumption and the conditional independence assumption, can give rise to computational efficiency but it comes with expense in accuracy. In probabilistic retrieval model, an ideal situation is to discard any independence assumption, even though it is almost mathematically intractable to do so. Therefore, we have to consider the trade-off between computational efficiency and accuracy. In practice, we can expect to achieve certain degree of accuracy of relevance probability estimation without significantly increasing the computational complexity. This can be accomplished by using (i) as few independence assumptions as possible and (ii) a *correlation function*[2] to identify any significant correlation relation among index terms so that only the significant ones are included in the probability calculation.

In this thesis, we propose a probabilistic retrieval model that is suitable for categorizing multiple-record Web documents based on logistic regression analysis and application ontologies. A document is a multiple-record document if it contains a sequence of similar chunks of information about the main entity of an ontology [ECJ+99]. We notice that many Web documents, such as advertisements, obituaries, etc., belong to this category. We choose application ontologies because they describe particular data of interest and capture the semantic relationships among data objects (index

[2] A correlation function is used to study the dependent relationship between two quantities and is usually defined as the product of the two quantities.

terms) in a real-world application using a simple conceptual modeling approach. In our approach, we adopt logistic regression analysis to study, i.e., curve-fit, (i) the relationships among relevance probabilities and term frequencies of index terms, (ii) the density value, and (iii) the grouping value of documents in a training set. We consider applying probabilistic model for categorizing a collection of documents into two non-overlapping sets according to an application ontology. Documents in one set are relevant to the application ontology, whereas documents in the other are not. In this binary-categorization approach, we determine the relevance probabilities based on imperfect knowledge about documents. We first study (i.e., curve-fit) the statistical relationships between the relevance probabilities of documents in a training set with respect to an application ontology and the term frequencies of document properties, e.g., term frequencies of index terms. Hereafter, the relevance probability of a test document is interpolated from the fitting-curves. (Normally, a probability curve, which is a function of the term frequency of an index term, looks like an "S", and hence is called an *S-curve.*) In addition, we adopt (1) the *density heuristic*, which measures the percentage of the content in a document applicable to the application ontology, and (2) the *grouping heuristic*, which measures the degree of structural similarity between each "record" in a document and the anticipated application ontology.

In order to have a robust probabilistic retrieval model, we collect the training and test documents that are both "narrow" and "broad" in content. The content should be *broad* because we want our probabilistic model to handle different kinds of documents. The content should be *narrow* because we would like to catch subtle differences between similar documents. The performance of our probabilistic model is evaluated by calculating the *precision*, *recall*, and *accuracy* ratios using our probabilistic model on a collection of test documents. Recall is the fraction of the relevant documents which has been retrieved, precision is the fraction of the retrieved documents that are relevant, and accuracy is defined as the ratio of the total number of documents minus the false drops and false positives to the total number of documents [BYRN99].

This thesis is organized as follows. In Chapter 2, we discuss application ontologies, logistic regression analysis, probabilistic retrieval model, and the probabilistic retrieval model based on logistic regression. In Chapter 3, we use the car-ads application ontology as an example to illustrate how we apply the probabilistic retrieval model based on logistic regression on a real world application. In Chapter 4, we present the experimental results on both car-ads and obituary ontologies. In the lase chapter, we conclude the thesis work.

# Chapter 2

# Preliminaries

## 2.1 Application Ontology

In this thesis, we propose a probabilistic retrieval model based on logistic regression and application ontologies for classifying multiple-record Web documents. An application ontology consists of two components:

1. a *conceptual model instance*, which describes sets of objects, sets of relationships among objects, and constraints over objects and relationships, and

2. *data frames*, which define the lexical appearance of constant objects within each object set and appropriate keywords that are likely to appear in a document when constant objects in the object set are mentioned [EJN99].

Figure 2.1 shows a *graphical* representation of the object sets and relationship sets of a car-ads application ontology, whereas Figure 2.2 shows a portion of the *textual* representation of the car-ads ontology, which includes all object and relationship sets, cardinality constraints (lines 1-8), and a few lines of its data frames (lines 9-41). An object set in an application ontology represents a set of objects which is

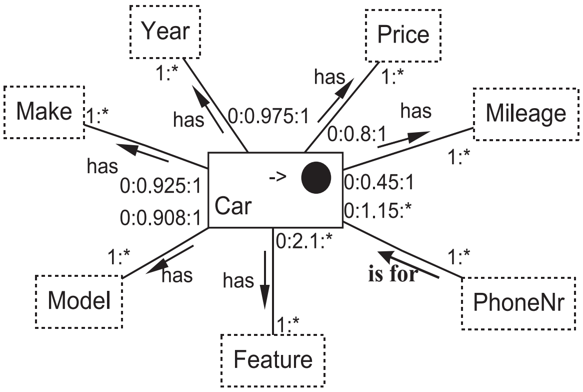



Figure 2.1: Car-ads ontology—graphical

```
 1. Car [-> object];
 2. Car [0:0.975:1] has Year [1:*];
 3. Car [0:0.925:1] has Make [1:*];
 4. Car [0:0.908:1] has Model [1:*];
 5. Car [0:0.45:1] has Mileage [1:*];
 6. Car [0:2.1:*] has Feature [1:*];
 7. Car [0:0.8:1] has Price [1:*];
 8. PhoneNr [1:*] is for Car [1:1.15:*];
 9. Year matches [4]
10.    constant { extract "\d{2}";
11.               context "([^\$\d]|^)[4-9]\d[^,\dkK]";
12.               substitute "^" -> "19";
13.                },
14.              { extract "\d{2}";
15.               context "([^\$\d]|^)[4-9]\d,[^\d]";
16.               substitute "^" -> "19";
17.               },
18.              { extract "\d{2}";
19.               context "\b'[4-9]\d\b";
20.               substitute "^" -> "19";
21.               }, 22.          { extract "\d{4}";
23.               context "([^\$\d]|^)(19[4-9]\d|200\d)[^,\dkK]";
24.                },
25.              { extract "\d{4}";
26.               context "([^\$\d]|^)(19[4-9]\d|200\d),[^\d]";
27.               };
28. end;
29. Make matches [10]
30.    constant { extract "\bacura\b"; },
31.              { extract "\balfa(\s*romeo)?\b"; },
32.              { extract "\bamc\b"; },
33.              { extract "\bam\s*general\b"; },
34.              { extract "\baudi\b"; },
35.              ...
36. end;
37. ...
38. Mileage matches [8]
39. ...
40.   keyword  "\bmiles\b", "\bmi\.", "\bmi\b", "\bmileage\b";
41. end;
42. ...}
```

Figure 2.2: Car-ads ontology—textual

either *lexical* or *nonlexical*. In our probabilistic model, lexical objects are treated as index terms. Lexical object sets are shown in the graphic representation as dashed boxes (e.g., **Make** and **Model** in Figure 2.1) or *data frames* of object sets with a *constant* declaration in the textual representation (see line 10 or 30 in Figure 2.2 for an example) of an application ontology. *Solid boxes* or *data frames* of object sets without *a* constant declaration are nonlexical object sets.

An application ontology may have more than one nonlexical object set. (In the car-ads ontology, the only nonlexical object set is *Car*, but in a more complex ontology, such as obituary, there can be more than one nonlexical object set.) One of the nonlexical object sets is called the *object of interest*. The graphical notation "$\longrightarrow \bullet$" (see **Car** in Figure 2.1 for an example) or the textual notation "[$\longrightarrow$ object]" (see line 1 in Figure 2.2 for an example) denotes the object of interest. Only one object of interest may appear in an application ontology.

A line connecting boxes in a graphical representation of an application ontology denotes a relationship set among the linked object sets. A labeled reading-direction arrow next to a relationship-set line, along with the names of the connected object sets, yields the name of the relationship set. **Car** *has* **Model** in Figure 2.1, for example, names the relationship between **Car** and **Model**. In the textual representation, we use this name to represent both the relationship set and the connected object sets. For example, line 4 in Figure 2.2 defines the relationship set between **Car** and **Model**.

The *min:max* or *min:ave:max* constraint specified next to the connection between an object set and a relationship set in a graphical representation is the *participation constraint* of the object set in the relationship set. *Min, ave,* and *max* denote the minimum, average, and maximum number of times an object in an object set can, or is expected to, participate in a relationship set, respectively, whereas * designates an unknown but finite maximum number of times an object in an object set can participate in a relationship set. In the textual representation for the car-ads ontology, the participation constraints are listed from line 2 to line 8. The participation constraint on **Car** *has* **Year** i.e., [0:0.975:1] in the car-ads ontology as shown is Figures 2.1 and 2.2, specifies that a car advertisement should have no more than one occurrence of the manufacture year, but it could have none, and on the average, it appears 0.975 times in a car advertisement.

## 2.2 Regression Analysis

Regression analysis is a statistical tool used to study the *relation* between a *dependent* (response) variable and a set of *independent* (explanatory) variables so that the value of the dependent variable can be predicted from the values of the independent ones [NWH83]. Each dependent and independent pair is called an *observation*. A scatter plot including many observation points may reveal a functional relation between the dependent and independent variables. There are two different kinds of relations among variables: functional relations and statistical relations.

### 2.2.1 Functional Relations

A *functional relation* can be expressed as a perfect function of the form

$$y = f(\vec{x}), \quad \vec{x} = (x_1, x_2, \cdots, x_n) \tag{2.1}$$

where $\vec{x}$ is a set of independent variables and $y$ is a dependent variable. The simplest functional relation is the linear function $y = C_0 + C_1 x$ between two variables, where $x$ is an independent variable and $y$ is a dependent variable. Given a particular value of $x$, this linear function determines the corresponding value of $y$. For example, given the linear function $y = 1 + 2x$, the pairs, i.e., observations, $(y, x) = (3, 1), (5, 2), (7, 3), \cdots$ etc., can be generated. We draw these observation points on a $X$-$Y$ plot as illustrated in Figure 2.3. As shown in the figure, all the observation points fall directly on the line defined by the function. Thus, we say that there is a functional relation between $x$ and $y$, and the function is in the form of $y = 1 + 2x$.

### 2.2.2 Statistical Relations

Unlike a functional relation, a *statistical relation* does not show a perfect relation between the dependent and independent variables, which means that some of the observation points do not fall directly onto the curve of a well-defined function. In reality, most relations among variables that we encounter are not perfect. Often, they are statistical, instead of functional, relations among variables. This concept can be illustrated using a uni-variable case (i.e., the case with only one independent variable), which can easily be extended to a multiple-variable situation. For example,

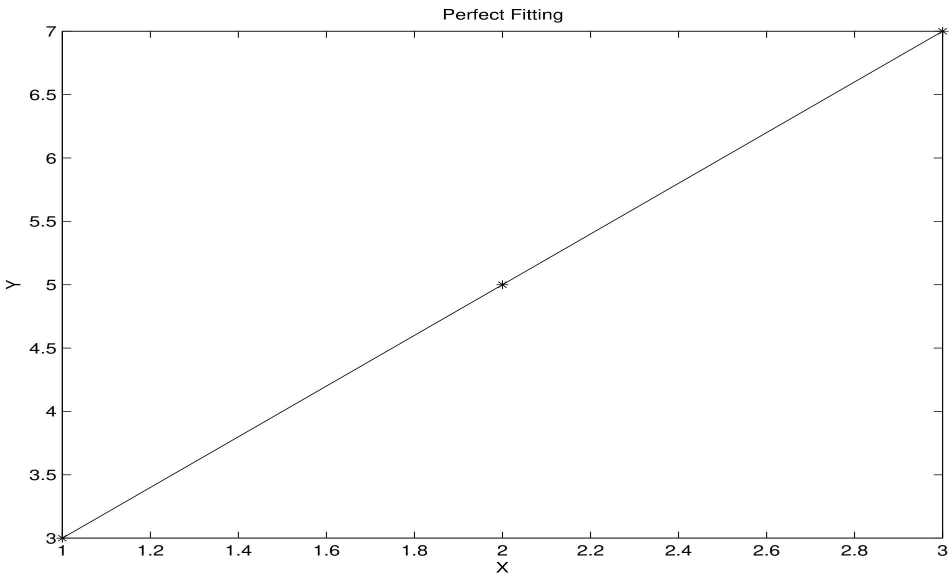



Figure 2.3: The functional relation $y = 1 + 2x$

the relation between the voltage $V$ crossing a resistor $R$ and the current $A$ passing through it. For several measured observations in this relation, each observation $(V, A)$ represents a point on a $V$-$A$ graph as illustrated in Figure 2.4.

According to the Ohm's law, for a pure resistor, $V$ and $A$ are linearly related through the resistance $R$ of the resistor. All observation points should fall directly on a straight line defined by $V = R \cdot A$. However, due to the systematic error of the measuring instruments (voltage and current meters), the intrinsic inductance, the capacitance of the resistor, and the temperature dependent resistance of the resistor, some observation points fall outside the straight line even though the points clearly indicate a linear relationship. This situation raises several important questions, which include: (i) "Can we find a function that best represents this statistical relation?",

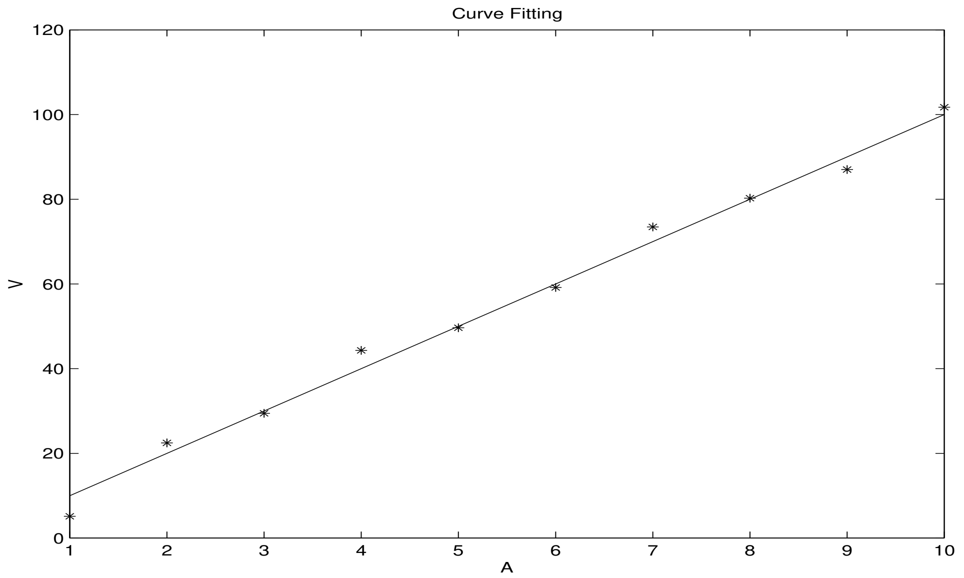



Figure 2.4: Statistical relation between voltage $V$ and current $A$

and (ii) "How good is the best? What are the criteria for this measurement?" The answer to the first question is "yes." We can use a method called *curve-fitting*, which adopts a well-defined function to represent a statistical relation. We can always find such a functional relation, which at least, can describe the trend of a dependent variable with respect to the change of independent variables. The answer to the second question is not as simple as the first one. There are different ways to fit statistical relations, and the criteria for the "best fit" differ from one model to another. In the linear regression model, a straight line is used to fit the points, whereas in logistic regression, an exponential function is used. In physics, the fitting curve can take any functional form. Therefore, the exact functional form is problem-dependent, i.e., it depends on the point-distribution and reasonable physical explanations explicitly contained in the chosen functional form. In case of a uni-variable linear relation, this fitting function is expressed as

$$\hat{y} = C_0 + C_1 x \tag{2.2}$$

where $\hat{y}$ is a functional estimation of observation $y$. There are several ways to evaluate the unknown regression coefficients $C_0$ and $C_1$ of the curve-fitting. The most commonly used approach adopts the *least squares criterion*, which makes use of a list of observations $(y_i, x_i),\ i = 1, 2, \cdots, n$, and a fitting function $\hat{y_i} = f(x_i)$ to define the function $\sigma = \sum_{i=1}^{n}(y_i - \hat{y_i})^2$. The summation $\sigma$ is called *residual sum of squares*. The curve with the best fit under the least squares criterion is the one that minimizes this summation.

To compute an estimation of the unknown coefficients $C_0$ and $C_1$, a linear regression analysis function takes the observations $(y_i,\ x_i),\ i = 1, 2, \cdots, n$, from the sample data (training set) as input. The coefficient $C_1$ is the slope of the regression line, which indicates the change of the dependent variable $y$ per unit change of the independent variable $x$. Thus, the sign and the magnitude of $C_1$ show how and how much the independent variable will influence the dependent variable. The coefficient $C_0$ is the intercept of the regression line and usually does not have any particular meaning in regression analysis. In the case of multiple-variable linear regression, the fitting function is $\hat{y} = C_0 + C_1 x_1 + \cdots + C_n x_n$, and the input is a list of data sets $(y_i, x_{1i}, \cdots, x_{ni}),\ i = 1, 2, \cdots, m$, and the list of unknown coefficients are $C_0, C_1, \cdots,$ and $C_n$.

A linear regression model fits all kinds of data using a straight line. However, in many situations, a functional relation might be *curvilinear* instead of linear. Figure 2.5 shows a curvilinear functional relation between a dependent variable and an independent variable. Obviously, a straight line will not be a good fit under this circumstance. The common approach to deal with a curvilinear relation is by data transformation. In Figure 2.6, the same set of data used in Figure 2.5 is plotted with the independent variable $x$ transformed as $x' = x^2$, where $x'$ is the transformed variable. With the transformed data, the scatter plot in Figure 2.6 shows a reasonable linear relationship, and thus a simple linear regression can be applied. Note that quadratic, logarithms, and inverted functions are commonly used transformations. Often a transformation for an independent variable, dependent variable, or both is sufficient to transform a curvilinear relation into a simple linear one [DWHL89].

## 2.3 Logistic Regression Analysis

In our probabilistic information retrieval model, we want to study the statistical relation between the relevance probability of a document with respect to an application ontology and the document properties, i.e., term frequencies of index terms appeared in the document [CLvRC98, Fuh92]. A relevance probability is influenced by document properties and can be treated as a dependent variable, whereas document properties can be treated as independent variables.

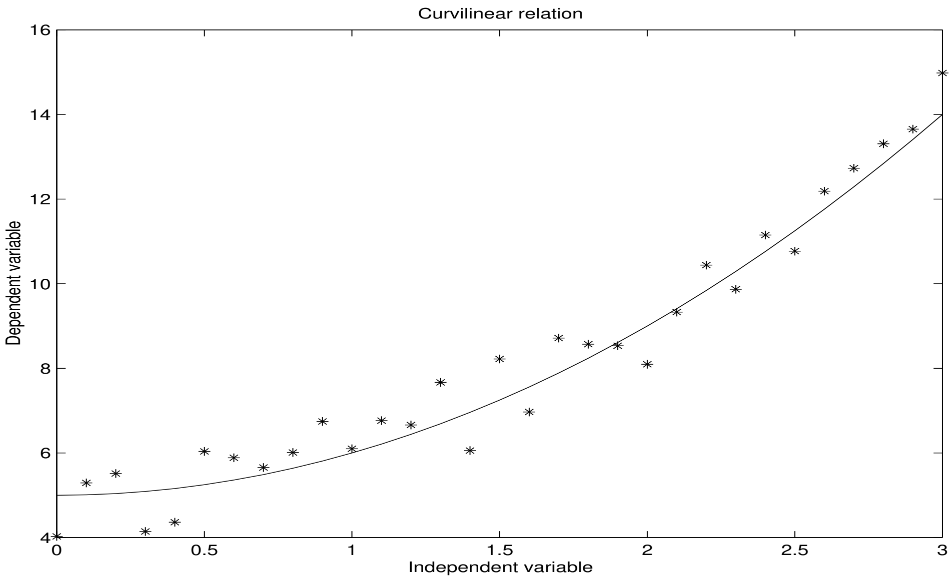



Figure 2.5: Curvilinear relation between a dependent variable and an independent variable

### 2.3.1 Single $S$-Curve Distribution

In a training document set, the relevance probability of a document to an application ontology is dichotomous. This kind of dichotomous behavior is quite different from a linear regression model in which a dependent variable can take on many possible values (see the differences between Figures 2.6 and 2.7). A simple illustration in Figure 2.7 shows why an $S$-curve is more appropriate to fit the data than a straight line in this circumstance. Figure 2.7 shows the distribution of relevance probability $P$ of a group of training documents applied to the car-ads application ontology for the frequencies of the index term **Make**. Each '*' on the graph represents a <probability, term-frequency> pair. For this graph a straight line would cut across both the $P = 0$

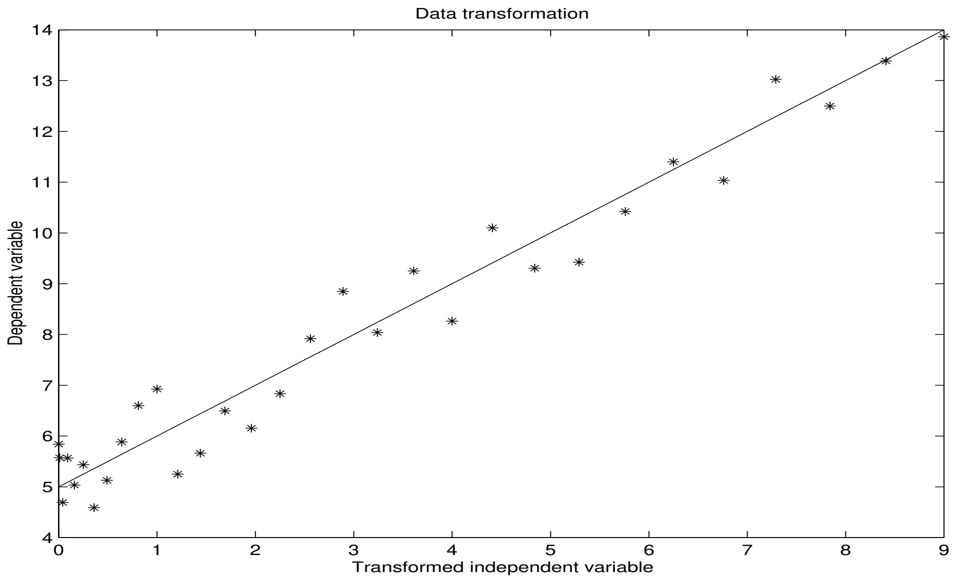



Figure 2.6: Converted from a curvilinear relation of Figure 2.5 to a linear one by data transformation

and $P = 1$ lines with most of the points located away from the line. Obviously, an $S$-curve is a better fit than a straight line in this case. There are several choices for the $S$-curve, and the commonly used one is

$$y = \frac{1}{1 + \exp\left(-(C_0 + C_1 x_1 + \cdots + C_n x_n)\right)} \tag{2.3}$$

where $y$ is a dependent variable and $x_1, \cdots, x_n$ are independent variables. The unknown coefficients $C_0, C_1, \cdots, C_n$, as in the linear regression model, are called *regression coefficients*. A regression model, which uses Equation 2.3 as the fitting curve, is called a *logistic regression model*.

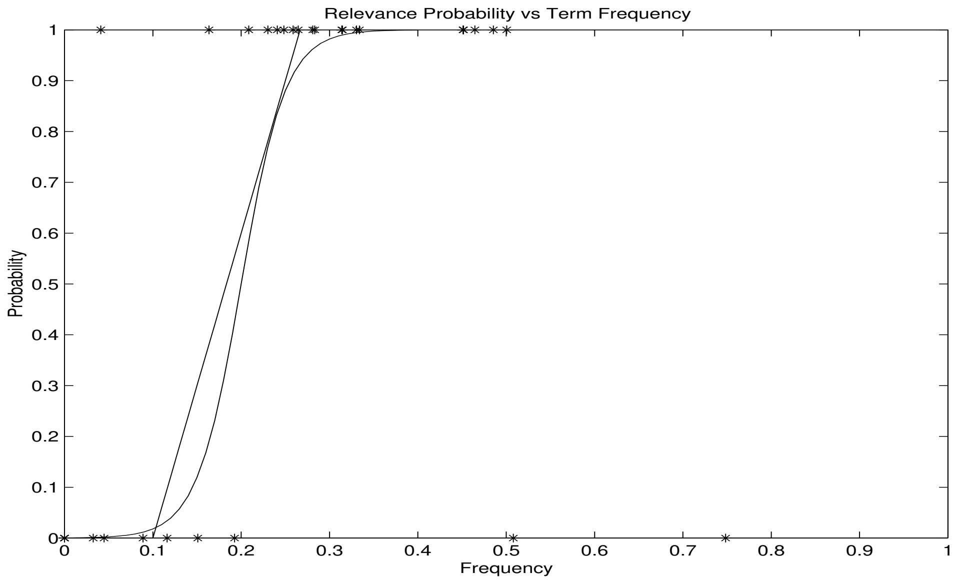



Figure 2.7: Logistic versus linear regression on the normalized term frequency of the index term **Make** in the car-ads application ontology

### 2.3.2 The Likelihood Function

The general approach for estimating the unknown coefficients $C_0, C_1, \cdots, C_n$ in the logistic regression model is called *maximum likelihood* [DWHL89]. The likelihood is evaluated by a *likelihood function*, which is defined as follows: given an observation point $(y_i, x_i)$, if the conditional probability that $y_i = 1$ is $P(y_i = 1|x_i)$, then the probability that $y_i = 0$ is $1 - P(y_i = 1|x_i)$. The probability that we actually observe on $(y_i, x_i)$ is then given by [DWHL89] as follows:

$$l'(y_i, x_i) = P(y_i = 1|x_i)^{y_i} \times [1 - P(y_i = 1|x_i)]^{1-y_i}. \tag{2.4}$$

This probability is called the *likelihood of observing the point* $(y_i, x_i)$. For information retrieval using logistic regression, the dependent variable should be the relevance probability. Thus, we replace $y$ in Equation 2.3 by probability $P$. For an observation point $(P_i, x_i)$, Equation 2.3 becomes

$$\hat{P}(x_i) = \frac{1}{1 + \exp\left(-(C_0 + C_1 x_i)\right)} \tag{2.5}$$

where $\hat{P}(x_i)$ is the function estimation of $P_i$. For each observation point $(P_i, x_i)$, its contribution to the likelihood is calculated by the term [DWHL89]

$$l'(P_i, x_i) = \hat{P}(x_i)^{P_i}[1 - \hat{P}(x_i)]^{1-P_i}. \tag{2.6}$$

We assume that all observations are independent events[1], and hence the likelihood function $L$ is the product of all the $l'(P_i, x_i)$s from all observations, i.e.,

$$L = \prod_{i=1}^{n} l'(P_i, x_i)\ . \tag{2.7}$$

Logistic regression is used to solve for unknown regression coefficients that maximize $L$. This is done by first differentiating $L$ with respect to the regression coefficients and then setting the resulting differential equations to zero. If $L$ does have a maximum, the regression coefficients can be determined from these differential equations [DWHL89].

### 2.3.3 Significant Independent Variables

Given a chosen independent variable, we need to assess its significance in the determination of the corresponding dependent variable. In logistic regression, the significance

[1] An example of an independent event is that if we randomly select a ball from a collection of five black balls and five white balls and then put it back, the next random selection is independent of the previous drawing.

of a chosen independent variable $v$ is evaluated by comparing the likelihood functions with and without $v$. A quantity $G$ of $v$ is defined as

$$G = -2\ln\left[\frac{\text{likelihood without } v}{\text{likelihood with } v}\right]. \tag{2.8}$$

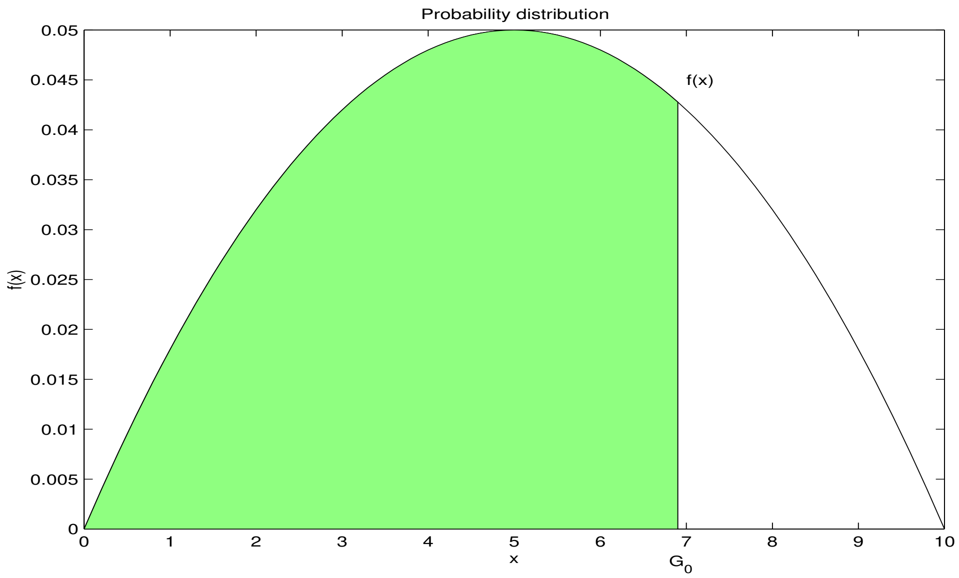



Figure 2.8: Probability distribution of quantity $G$

According to the definition of the likelihood function, a larger value of the likelihood function corresponds to a better fit, i.e., the larger the quantity $G$ is, the more significant $v$ is. Even though there is no absolute criterion on how large $G$ should be, the significance is evaluated by calculating an associated $p$-value of $G$. The $p$-value of $G$ is the probability of observing a value smaller than $G$. $G$ follows a *chi-square distribution* with its probability distribution function $f(x)$ illustrated in Figure 2.8.

The probability that $G \leq G_0$ is given by

$$p(G \leq G_0) = \int_0^{G_0} f(x)dx \tag{2.9}$$

This integral corresponds to the shaded area under the probability distribution $f(x)$ in Figure 2.8. From the training data, we can calculate the value of $G_0$ associated with an independent variable. If the probability of $G \leq G_0$ is above a *confidence level* (say 95%), we say that the independent variable is significant. In most logistic regression packages and logistic regression references, a *risk factor* $f$, which is defined as one minus the confidence level, is used to indicate the significance of the independent variable. Hence, if the confidence level is 95%, the risk factor is 1 - 0.95 = 0.05. The quantity $f$ is associated with the probability of $G > G_0$. Hence, if a $p$-value is smaller than the risk factor, the corresponding variable is categorized as significance.

Discerning significant and insignificant independent variables is a critical task in regression analysis. Initially, we would like to include as many independent variables as possible because we do not know which one is significant, and we are afraid of missing any of them. Obviously, some of these variables are more significant than the others. Including too many insignificant independent variables in a model, however, causes *over-fitting*, which causes the model to lose its ability to generalize and will deteriorate. There is an extreme case when the presence of an independent variable contributes little or nothing to the discrimination of two discrete states, i.e., relevant or irrelevant. Under this circumstance, the logistic regression analysis will not converge, and no coefficients and statistical information, other than the insignificance of the variable, can be concluded.

## 2.4 Probabilistic Retrieval Model

A probabilistic retrieval model solves an IR problem based on probability theory. For a specific query, a document collection can be partitioned into two non-overlapping sets: documents in one set are relevant to the query, whereas documents in the other are not. Without reading through a document in the collection, how can we determine its relevancy to the given query? Even though we are given a set of summary data (e.g., index terms and their corresponding term frequencies) about the document as evidence, we still cannot guarantee to have an accurate classification of the document with respect to the query. We have to guess its relevancy based on the imperfect knowledge about the document. An approach to guess is by estimating a relevance probability $P_q(R|d)$, where $R$ denotes the relevancy of document $d$ with respect to the specific user query $q$, which is represented by an application ontology in our probabilistic retrieval model. $P_q(R|d)$ is read as: "the relevance probability of a document $d$ with respect to a given query $q$." To simplify the notation and assuming that the query being referred to is $q$, we can drop the subscript $q$ and rewrite the relevance probability symbol as $P(R|d)$. In order to estimate $P_q(R|d)$, we make the following assumptions:

1. The relevance of $d$ to $q$ depends only on $d$ and $q$, i.e., the relevance of $d$ to $q$ is independent of other documents in the collection to which $d$ belongs.

2. Given an application ontology, $d$ in the collection can be described by:

   (a) a set of <index term : term frequency> pairs: $< A_1 : x_1 >, \cdots, < A_n : x_n >$, $n \geq 1$,

   (b) a *density heuristic* value, which measures the percentage of the matched charactors in $d$ with respect to $q$. We represent this value by the symbol $y$, and

   (c) a *grouping heuristic* value, which measures the degree of structural similarity between each "record" in $d$ and $q$. We represent this value by the symbol $z$.

The first assumption simply states the fact that each document retrieved is an independent event, i.e., the relevance of documents determined earlier will not affect the relevance of the others to be determined later. Part (a) in the second assumption states that part of our view of $d$ is a set of term frequencies $(x_1, \cdots, x_n)$, where the ordering of the term frequencies is assumed to be the ordering of $A_1, \cdots, A_n$, the corresponding index terms. Parts (b) and (c), along with part (a), determine the relevant probability of $d$ with respect to $q$. Hence, the relevance probability of $d$ can be written as $P(R|(x_1, \cdots, x_n, y, z))$. In our proposed probabilistic retrieval model, $P(R|(x_1, \cdots, x_n, y, z))$ is first broken down into $P(R|x_1)$, $\cdots$, $P(R|x_n)$, $P(R|y)$, and $P(R|z)$ using an independence assumption. Then the logistic regression analysis is applied on each probability term. Thereafter, dependent relationships among index terms can be studied and included in relevance probability estimation to compensate for any bias caused by the independent assumption.

## 2.5 Relevance Assumptions in Probabilistic Retrieval

In probabilistic retrieval, it is computationally inefficient to estimate the relevance probability of a document with respect to a user query [CLvRC98], especially when the index terms used in the model are not mutually independent. The relevance probability estimation of a document with respect to a user query usually involves many other probability calculations, such that the total number of probabilities involved, and their complexity are determined by the complexity of dependence relations among index terms in the document. If no knowledge or assumption about dependent relations is available, we have to consider all possibilities. The total number of dependence relations among index terms is an exponential function on the number of index terms that appear in the document. This could involve a tremendous amount of computational time. (Whenever we speak of dependence, we refer to *stochastic dependence*, which is the relation implicitly contained in a family of random variables. Unlike a logical dependence which can be explicitly inferred using logical deduction, i.e., $A \longrightarrow B$ and $B \longrightarrow C$ imply $A \longrightarrow C$, a stochastic dependence is usually studied by a correlation function.)

### 2.5.1 Independence Assumptions

Early probabilistic models, such as the *binary independence indexing model* [vR75] and the *binary independence retrieval model* [vR75], use the mutual independence assumption exclusively to make models computationally workable, which by no means provides any justification to do so and inevitably cause distortion. Many probabilistic retrieval models [CLvRC98] use both the *absolute independence assumption* and the *conditional independence assumption* among chosen index terms. A simple probability manipulation shows that a combination of the *absolute independence assumption* and the *conditional independence assumption* can lead to a conclusion that contradicts the laws of the probability theory [CLvRC98]. For example, given an absolute independence assumption

$$P(a, b) = P(a) \cdot P(b) \tag{2.10}$$

and a conditional independence assumption

$$P(a, b|R) = P(a|R) \cdot P(b|R) \tag{2.11}$$

where $a$ and $b$ are document properties, e.g., the frequency of an index term, and $R$ indicates that the corresponding document is relevant to a given query. $P(a, b)$ denotes the probability that properties $a$ and $b$ are both present simultaneously, which is also called the *joint probability* of $a$ and $b$, whereas $P(a, b|R)$ denotes the probability of both $a$ and $b$ present if a document is relevant to a user query. If $a$ and $b$ are strongly interdependent such that they coexist, then

$$P(a, b, R) = P(a, R) = P(b, R) \tag{2.12}$$

Without loss of generality, we assume that $P(a, b, R) < P(R)$, where $P(R)$ is the probability that a document is relevant to a user query. By using the probability theory (see Appendix A.1) and the independent assumptions, $P(a, b, R)$ can be expressed as

$$\begin{aligned} P(a, b, R) &= P(a, b|R) \cdot P(R) \\ &= P(a|R) \cdot P(b|R) \cdot P(R) \\ &= \frac{P(a|R) \cdot P(R) \cdot P(b|R) \cdot P(R)}{P(R)}. \end{aligned} \tag{2.13}$$

Since $P(a, b) = P(a|b) \cdot P(b)$ (see Appendix A.1),

$$\begin{aligned} P(a, b, R) &= \frac{P(a, R) \cdot P(b, R)}{P(R)} \\ &= \frac{P(a, b, R)^2}{P(R)} \end{aligned} \tag{2.14}$$

and hence $P(a, b, R) = P(R)$. This conclusion contradicts the assumption that $P(a, b, R) < P(R)$. This inconsistency is caused by the independence assumption. Based on this argument, it is desirable to remove or at least weaken the independence assumption when we construct a probability retrieval model.

### 2.5.2 Linked Independence Assumption

In order to reduce the distortion introduced by using the mutual independence assumption, i.e., the binary independence assumption, we adopt the *Linked Dependence* model [CGD92]. In the Linked Dependence assumption, the statistical dependence among the index terms is assumed to have a constant magnitude $K$, and the probabilities $P(A,B|R)$ and $P(A,B|\bar{R})$ can be broken down as

$$\begin{aligned} P(A,B|R) &= K \cdot P(A|R) \cdot P(B|R), \quad \text{and} \\ P(A,B|\bar{R}) &= K \cdot P(A|\bar{R}) \cdot P(B|\bar{R}) \end{aligned} \tag{2.15}$$

where $K$ is any positive number, and $\bar{R}$ denotes irrelevance of a document with respect to a query. The role of $K$, though still debatable, can be interpreted as a *crude measure* of the statistical dependency between $A$ and $B$ [CGD92]. (If $K = 1$, the model converges to the binary independence model.) The *Linked Dependence* assumption can be extended to

$$\begin{aligned} P(x_1, \cdots, x_n|R) &= K \cdot P(x_1|R) \cdots P(x_n|R) = K \cdot \prod_{i=1}^{n} P(x_i|R), \quad \text{and} \\ P(x_1, \cdots, x_n|\bar{R}) &= K \cdot P(x_1|\bar{R}) \cdots P(x_n|\bar{R}) = K \cdot \prod_{i=1}^{n} P(x_i|\bar{R}) \end{aligned} \tag{2.16}$$

where $n$ is the number of index terms. It is easy to prove that, under the extreme case where $a$ and $b$ are strongly interdependent, even the Linked Dependence model can inflate probability and cause distortion [CGD92]. Suppose $a$ and $b$ are strongly interdependent such that they coexist, then

$$\begin{aligned} &P(a,b|R) = P(a|R) = P(b|R) \\ &P(a,b|\bar{R}) = P(a|\bar{R}) = P(b|\bar{R}) \end{aligned} \tag{2.17}$$

Using the Linked Dependence assumption, the odds of $P(a, b|R)$ is written as

$$\begin{aligned}\frac{P(a,b|R)}{P(a,b|\bar{R})} &= \frac{K \cdot P(a|R) \cdot P(b|R)}{K \cdot P(b|\bar{R}) \cdot P(b|\bar{R})} \\ &= \frac{P(a|R)}{P(a|\bar{R})} \cdot \frac{P(b|R)}{P(b|\bar{R})} \end{aligned} \tag{2.18}$$

By Equation 2.17, we have

$$\frac{P(a,b|R)}{P(a,b|\bar{R})} = \frac{P(a,b|R)^2}{P(a,b|\bar{R})^2}. \tag{2.19}$$

The solution of this equation is $\frac{P(a,b|R)}{P(a,b|\bar{R})} = 1$ or $\frac{P(a,b|R)}{P(a,b|\bar{R})} = 0$, which means that either $P(a, b|R) = 0$ or $P(a, b|R) = P(a, b|\bar{R})$. The assumption that both $a$ and $b$ coexist does not necessary lead to these conclusions. These conclusions are caused by the Linked Independence assumption and it could bias relevance probability calculation [CGD92]. For this reason, we will not rely on the Linked Independence assumption alone to address dependent relationships among index terms. Instead, we will use correlation functions to identify dependent relations and include only the important ones into our relevance probability calculation.

## 2.6 Odds of Occurrence in Logistic Regression Analysis

According to Equation 2.16, the relevant probability $P(x_1, \cdots, x_n, y, z|R)$ can be broken down into a simpler expression. However, the value of $K$ is unknown. In order to remove $K$ from the computation, we use a probability ratio, called the *odds of occurrence* of an event $E$, which is defined as $O(E) = \frac{P(E)}{P(\bar{E})}$. $O(E)$ is the ratio of the probability that an event $E$ occurs to the probability that E does not occur. Based on this definition and $P(x,R) = P(x|R) \cdot P(R) = P(R|x) \cdot P(x)$ and $P(x,\bar{R}) = P(x|\bar{R}) \cdot P(\bar{R}) = P(\bar{R}|x) \cdot P(x)$ (see details in Appendix A.1), the ratio of $P(x|R)$ and $P(x|\bar{R})$ can be expressed as

$$\begin{aligned}\frac{P(x|R)}{P(x|\bar{R})} &= \frac{P(x,R)/P(R)}{P(x,\bar{R})/P(\bar{R})} \\ &= \frac{P(R|x) \cdot P(x) \cdot P(\bar{R})}{P(\bar{R}|x) \cdot P(x) \cdot P(R)}.\end{aligned} \tag{2.20}$$

By the definition of *odds of occurrence*, $\frac{P(R|x)}{P(\bar{R}|x)} = O(R|x)$ and $\frac{P(R)}{P(\bar{R})} = O(R)$. Using these two equalities, Equation 2.20 can be simplified as

$$\frac{P(x|R)}{P(x|\bar{R})} = \frac{O(R|x)}{O(R)}. \tag{2.21}$$

Also, applying the Linked Dependence model, the following equation holds:

$$\begin{aligned}\frac{P(x_1,\cdots,x_n,y,z|R)}{P(x_1,\cdots,x_n,y,z|\bar{R})} &= \frac{K \cdot P(x_1|R)\cdots P(x_n|R) \cdot p(y|R) \cdot p(z|R)}{K \cdot P(x_1|\bar{R})\cdots P(x_n|\bar{R}) \cdot P(y|\bar{R}) \cdot P(z|\bar{R})} \\ &= \prod_{i=1}^{n} \frac{P(x_i|R)}{P(x_i|\bar{R})} \cdot \frac{P(y|R)}{P(y|\bar{R})} \cdot \frac{P(z|R)}{P(z|\bar{R})}.\end{aligned} \tag{2.22}$$

where $K$ is cancelled in the equation. Since each term in Equation 2.22 follows the equality given in Equation 2.21,

$$\frac{O(R|x_1,\cdots,x_n,y,z)}{O(R)} = \prod_{i=1}^{n} \frac{O(R|x_i)}{O(R)} \cdot \frac{O(R|y)}{O(R)} \cdot \frac{O(R|z)}{O(R)}. \tag{2.23}$$

Taking natural logarithm to both sides of Equation 2.23, the equation can be rewritten as

$$\ln(O(R|x_1, \cdots, x_n, y, z)) = \sum_{i=1}^{n} [\ln(O(R|x_i)) - \ln(O(R))] + \ln(O(R|y)) + \ln(O(R|z)) - \ln(O(R)) \tag{2.24}$$

where the mathematical expression $\ln(O(E))$ is called the *logodds of an event E.*

Each logodds term, except $ln(O(R))$, in Equation 2.24 is calculated by applying logistic regression on it. In logistic regression, each $P(R|x_i)$ is represented by a logistic function[2] as shown in Equation 2.25. A set of unknown parameters $C_0$ and $C_1$ are called *regression coefficients*, which can be estimated by using logistic regression, assuming that $x_i$ is an independent variables, i.e.,

$$P(R|x_i) = \frac{1}{1 + \exp\left(-(C_0 + C_1 x_i)\right)}. \tag{2.25}$$

Using the data in a training set, each $\langle x_i, P(R|x_i)\rangle$ pair is calculated and fed into the logistic regression software package. Afterwards, the corresponding regression coefficients $C_0$ and $C_1$ are computed and accompanied by the associated $p$-value, which can be used to analyze the significance of the corresponding index term. Taking the logodds of $P(R|x_i)$ in Equation 2.25, the equation becomes

$$\ln(O(R|x_i)) = C_0 + C_1 \cdot x_i. \tag{2.26}$$

where $C_1$ determines the change of $\ln[O(R|x_i)]$ per unit change of $x_i$. The value of $C_1$ reflects the importance of $x_i$ in determining the relevance probability of a document

[2]A logistic function is a *continuous function*, e.g., $y = \frac{1}{1+\exp\left(-(C_0+C_1 x)\right)}$, bounded between 0 and 1 with $y = 0$ and 1 as the asymptotes.

with respect to a query (or an application ontology in our probabilistic retrieval model).

For a given test document $d$, the value of $x_i$ for the index term $A_i$ is inserted into Equation 2.26 to evaluate $\ln(O(R|x_i))$. By definition, the quantity $O(R)$ is equal to $\frac{P(R)}{P(\bar{R})}$ where $P(R)$ is the probability that a randomly chosen document in a designated training document set is relevant to a user query (e.g., if we select 20 documents of car-ads and 20 documents of something else in the car-ads training document set, the $P(R)$ is equal to 50% and $ln(O(R))$ is then equal to zero.). The logodds for all index terms i.e., $\ln(O(R|x_i))$, $i = 1 \cdots, n$, $\ln(O(R|y))$, and $\ln(O(R|z))$ as well as $\ln(O(R))$, are then plugged into Equation 2.24 to obtain the logodds of the relevance probability of a given document $d$ whose document properties include $x_i$ $(1 \leq i \leq n)$, $y$, and $z$. Finally, the relevance probability of $d$ is obtained by converting the logodds of the relevance probability of $d$ using the definition of logodds. Assume that $\text{sum} = \ln(O(R|x_1, \cdots, x_n, y, z))$, by the definition of logodds,

$$\begin{aligned} \text{sum} &= \ln\left[\frac{P(R|x_1, \cdots, x_n, y, z)}{P(\bar{R}|x_1, \cdots, x_n, y, z)}\right] \\ &= \ln\left[\frac{P(R|x_1, \cdots, x_n, y, z)}{1 - P(R|x_1, \cdots, x_n, y, z)}\right]. \end{aligned} \tag{2.27}$$

Using the definition of natural logarithm, Equation 2.27 is rewritten as

$$\begin{aligned} e^{\text{sum}} &= \frac{P(R|x_1, \cdots, x_n, y, z)}{1 - P(R|x_1, \cdots, x_n, y, z)} \\ e^{\text{-sum}} &= \frac{1 - P(R|x_1, \cdots, x_n, y, z)}{P(R|x_1, \cdots, x_n, y, z)} = \frac{1}{P(R|x_1, \cdots, x_n, y, z)} - 1. \end{aligned} \tag{2.28}$$

Hence, the relevance probability of a document with respect to a user query is evaluated by

$$P(R|x_1, \cdots, x_n, y, z) = \frac{1}{1 + e^{\text{-sum}}}. \tag{2.29}$$

As mentioned before, we would like to maintain a good balance on "good" document representation, i.e., using all information appearing in a document while avoiding the problem of over-fitting. We start with all the lexical objects (i.e., index terms) along with the density value and the grouping value defined in an application ontology to represent the content of documents. However, some of the index terms may not be statistically important. We use the $p$-value associated with the term frequencies of the corresponding lexical object to determine which index term should be included in our probabilistic retrieval model since adding insignificant index terms can potentially cause over-fitting and deteriorate the performance of ours probabilistic retrieval model.

# Chapter 3

# Probabilistic Retrieval Based on Logistic Regression Analysis

In this chapter, we apply the proposed probabilistic retrieval model based on logistic regression analysis to multiple-record Web documents. We use the car-ads application ontology as an example to illustrate our binary-categorization approach to classifying the documents. Based on a designated application ontology, we produce a retrieval model by doing the following five steps.

1. We (i) collect training documents, (ii) extract the number of occurrences of index terms (i.e., lexical objects in the corresponding application ontology) in each document $d$, and (iii) calculate the density and grouping heuristic values of $d$. We then construct a term frequency vector using the number of occurrences of each index term in $d$ as a component. Hereafter, we normalize the vector

by the estimated number of "records" in $d$. The normalized vector, along with the corresponding density and grouping value vectors computed as shown in Section 3.1.1, represent $d$ in our probabilistic retrieval model.

2. We apply logistic regression on the components of the normalized term frequency vectors, the density value vector, and grouping value vector extracted from the training set documents to obtain the corresponding regression coefficients and $p$-values. In addition, we use the $p$-values to decide which of these quantities are significant and should be included in calculating the relevance probability of a document with respect to the corresponding application ontology. For those quantities with large $p$-values, we use scatter plots to study the corresponding data distribution more carefully. If the large $p$-value is caused by the random or invariant data distribution, the corresponding quantity should be excluded. On the other hand, if the large $p$-value is caused by a double $S$-curve distribution, the training document set is split into two subsets according to the term frequencies[1] of the corresponding index term in the training documents, and logistic regression is applied to each subset.

3. We collect a test document set and use the same approaches as stated in Step 1 to form a document vector for each test document. We then use the regression coefficients to calculate the logodds of index terms, the density value, and the

[1]Unless stated otherwise, from now on whenever we mention term frequencies, we mean normalized term frequencies.

grouping value of each test document $d$. The summation of the logodds is used to compute the relevance probability of $d$ with respect to an application ontology using the definition of the logodds.

4. We evaluate the model performance in terms of the recall, precision, and accuracy ratios according to the categorization of relevant and irrelevant in the test documents with respect to the application ontology determined by our model.

5. We study the impact of correlation relations among index terms on the performance of our probabilistic model. By adding important correlation relations in our model, we are able to improve the model performance. However, there is a trade-off between the gain of accuracy and extra computation as well as human interference.

For a given application ontology, Steps 1 to 5 only need to be performed once to finalize the retrieval model unless the ontology and/or the training document set is updated.

## 3.1 Data Extraction and Normalization

### 3.1.1 Data Extraction

Data used in logistic regression is extracted from each document $d$ in a training set and analyzed to yield the following information:

1. the number of occurrences of each index term in $d$,
2. the total number of characters in $d$ and the total number of characters that match index terms. These two values are used to compute the density heuristic value $y$ by the ratio of the latter to the former, and
3. the sum of distinct lexical values from all "records" in $d$ and the product of the number of "records" in $d$ and the expected number of distinct lexical values in a document. The ratio of the latter to the former yields the grouping heuristic value $z$ of $d$.

### 3.1.2 Normalization of index terms

For a multiple-record document, if all records are similar in structure and content, the density and grouping values are relatively insensitive to the document size, and these two values can be used directly in logistic regression without any normalization with respect to the document size. In general, the number of occurrences of an index term in a multiple-record document is closely related to the size of the document since the occurrences of the index term increases as the number of records increases. Thus, it does not make sense to compute the document relevancy solely based on the number of occurrences of index terms if documents have different sizes. For example, a car-ad document $d_1$ containing two records may have two occurrences of **Year**, whereas another document $d_2$ containing hundreds of records about something else may also include two occurrences of **Year**. Judging exclusively by the numbers of occurrences of index terms in $d_1$ or $d_2$ regardless the size of $d_1$ or $d_2$, we may not be able to draw any conclusion on which document is closely relevant to an application ontology. It is necessary to compare $d_1$ and $d_2$ on a per "record" basis instead. In order to remove the size dependency in a multiple-record document, we should *normalize* the number of occurrences of each index term in a document by the estimated number of records in the document.

Initially, we normalize term frequencies of index terms in a document by the number of recognized "records" in the document. The number of "records", however, cannot be reliably extracted even for a real multiple-record Web document since not all multiple-record documents have clear record boundaries and record separators. In addition, our training and test documents contain many Web documents that do not include multiple records, and our parser is not "smart" enough to recognize the differences and tries to segment the documents into multiple-"records." The number of "records" so collected is bogus. Thus, a different normalization approach is needed. Based on this observation, it is necessary to adopt a normalization approach that can tolerant mistakes made by the parser. We adopt an approach that normalizes data by estimating the number of "records" in a document.

In Chapter 2, we have stated that a document in our probabilistic model is represented by a set of $n$ different <index term : term frequency> pairs along with the density heuristic value and the grouping heuristic value extracted from the document. Based on this assumption, we consider the following heuristics where a Web document is *partially* represented by an $n$-dimensional vector and each component of the vector is the number of occurrences of the corresponding index term in the document.

**Heuristics 1:** The number of occurrences of index terms in a document can be represented in a one-dimensional array. This data array can be viewed as an $n$-dimensional vector that partially represents the document from which the composed data is extracted.

According to this heuristics, we can partially represent a document in a vector form. For example, if the numbers of occurrences of the index terms **Year**, **Make**, **Model**, **Mileage**, **Price**, **Feature**, and **Phone** in a multiple-record Web document $d$ are 22, 4, 12, 2, 14, 7, and 84, respectively, then the corresponding document vector of $d$ is

$$\vec{V} = (22, 4, 12, 2, 14, 7, 84).$$

This vector is normalized by the estimated number of records in $d$. The estimated number of records is calculated by dividing the length of the document vector by the length of the car-ads ontology vector $\vec{u}$. The construction of $\vec{u}$ is based on the average participation constraints determined empirically from a collection of documents. These documents were extracted from 10 different regions that cover 120 sites in the United States with 12 documents retrieved from each region. From each of these documents, twelve individual car-ads were selected using an interleaving mechanism described in [ENX01]. Then the average participation constraints were counted by hand. For the car-ads ontology, the corresponding $\vec{u}$ is

$$\vec{u} = (0.975, 0.925, 0.908, 0.450, 0.800, 2.100, 1.150)$$

where the order of the components in $\vec{u}$ is fixed by the order the index terms **Year, Make, Model, Mileage, Price, Features**, and **Phone**. Since the length of $\vec{u}$ is

$$|\vec{u}| = \sqrt{0.975^2 + 0.925^2 + 0.908^2 + 0.45^2 + 0.8^2 + 2.1^2 + 1.15^2} = 3.03,$$

and the length of $\vec{V}$ is

$$|\vec{V}| = \sqrt{22^2 + 4^2 + 12^2 + 2^2 + 14^2 + 7^2 + 84^2} = 89.16,$$

the vector normalized by the estimated number of records in the document is

$$\vec{V_{\text{norm}}} = \frac{(22, 4, 12, 2, 14, 7, 84)}{89.16/3.03} = (0.75, 0.13, 0.41, 0.07, 0.48, 0.24, 2.85)$$

$\vec{V_{\text{norm}}}$ can also be interpreted as the vector that has the same length as the corresponding ontology vector since $\frac{\vec{V}}{89.16}$ is a unit vector. In our probabilistic retrieval model, both training and test set documents are partially represented by their corresponding normalized vectors. Hence, given a document $d$, the normalized vector along with the density and grouping values of $d$ represent $d$ by which the relevancy of $d$ with respect to an application ontology is solely determined.

## 3.2 Data Analysis of Training Set Documents

Document vectors constructed from the training set documents form an $m \times (n+2)$ matrix, where $m$ is the number of documents in the training set and $n+2$ is the dimension of the normalized vector plus the density and grouping values in each training set document such that each row in the matrix represents the vector of the corresponding document in the training set. For each document in the training set, its relevancy with respect to the application ontology of interest is manually inspected and assigned the probability value of one, if it is deemed relevant, or zero if it is deemed irrelevant. Using the car-ads application ontology and its corresponding training set documents, these document vectors together with their density and grouping values are shown in Table 3.1.

The first column, i.e., **Probability**, in Table 3.1 indicates the probability distribution among the training documents. **Probability** plays the role of the dependent variable in logistic regression analysis. The remaining columns are term frequency distribution for all index terms, the density value, and the grouping values, which play the role of independent variables in logistic regression. Logistic regression is applied on each <dependent variable, independent variable> pair.

Before applying logistic regression on each dependent and independent variable pair, the probability distribution with respect to the term frequencies of each index term as well as the density and grouping values in a training document set is studied in order to determine any functional relationships between any dependent and independent variables. In logistic regression analysis, the functional relation is studied between a dependent variable (e.g., the relevance probability of a document with respect to an application ontology) and an independent variable (e.g., the term frequency of an index term). These functional relationships have nothing to do with dependent (or independent) relationships among index terms. This study can be performed by making a scatter plot. We consider four different types of data distribution: (i) single $S$-curve distribution, (ii) double $S$-curve distribution, (iii) random or invariant distribution, and (iv) complete separation.

### 3.2.1 Single $S$-curve Distribution

Often the relevance probability of a document $d$ with respect to a user query monotonically increases as the term frequency of an index term in $d$ increases. In this case, the probability distribution with respect to the term frequency of an index term looks like an $S$, called an $S$-curve. Figure 3.1 shows a plot of relevance probability distribution versus the normalized term frequency of the index term **Make** extracted from the training document set of car-ads ontology. The horizontal axis includes the normalized term frequencies of the index term **Make**, whereas the vertical indicates the corresponding probability values. Since the plot is for the training documents, the only possible value of the relevance probability for each document is either one or zero. The scatter plot clearly shows the dependence of the relevance probability on the normalized term frequency values of **Make**. When the term frequency value is low (high, respectively), the corresponding document is more likely to be irrelevant (relevant, respectively). Note that even though this dependent relation is not perfectly monotonic, a single $S$-curve is the best fit for this distribution. (In logistic regression, a single $S$-curve indicates that a single regression analysis can be carried out for the entire data value domain.) In Figure 3.1, we observe that the normalized term frequencies of the set of documents with Probability = 1 overlap with that of the set of documents with Probability = 0. In logistic regression this overlapping is desirable since only with overlapping the likelihood function has a maximum and thus leads to convergence of regression coefficients.

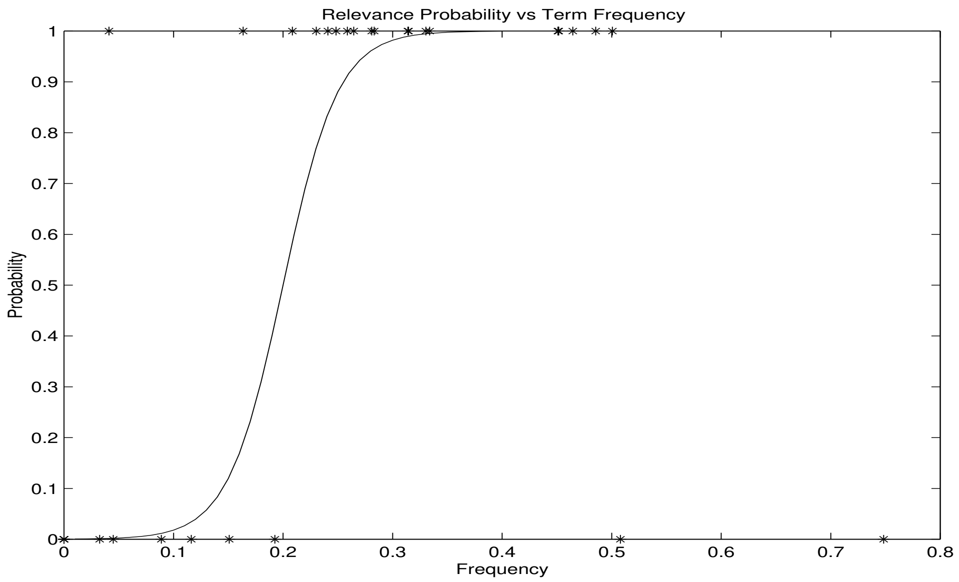



Figure 3.1: Scatter plot for the normalized term frequencies of the index term **Make** in car-ads ontology using the training set documents

### 3.2.2 Double $S$-curve Distribution

Sometimes, the functional relationship between the relevance probability and term frequency of an index term in a training document set is more complicated than a single $S$-curve implies. For example, it is possible that if an index term appears either infrequently or too often in a document, the document is probably irrelevant. Only when the term frequencies approximate the anticipated frequencies as specified in an application ontology, is the document probably relevant to the ontology. Without making a scatter plot, it is not easy to predict if the term frequency distribution of an index term is a single or double S-curve pattern since it depends on the chosen index term as well as the training document set. For example, in our car-ads application ontology, the term frequency of the index term **Make** has a single $S$-curve distribution. However, if we purposely choose some non-car-ads documents with many occurrences of car makers (e.g., an annual car survey), the frequency distribution of the index term **Make** then would show a double $S$-curve pattern. The double $S$-curve distribution, which consists of an $S$-curve and an upside down $S$-curve, is called a double $S$-curve. In the car-ads application ontology, this situation occurs to the index term **Year**. We can verify this assumption by examining Web documents which are indeed car-ads documents. For a car listed for sale, the seller usually lists the manufacture year once in the ads. If the parser recognizes more than one **Year** in the majority of the records, it is likely that the corresponding document is not a car-ads. Figure 3.2 shows the scatter plot for the index term **Year**. The data distribution clearly indicates a double

$S$-curve pattern. For this kind of distribution pattern, we apply logistic regression on two different term frequency ranges. For example, in the car-ads application, we select a threshold value of 0.39 that splits the data points, i.e., term frequencies of **Year** in the training set, in half. We apply two logistic regression analyses on the term frequency intervals [0, 0.39) and [0.39, 1], respectively.

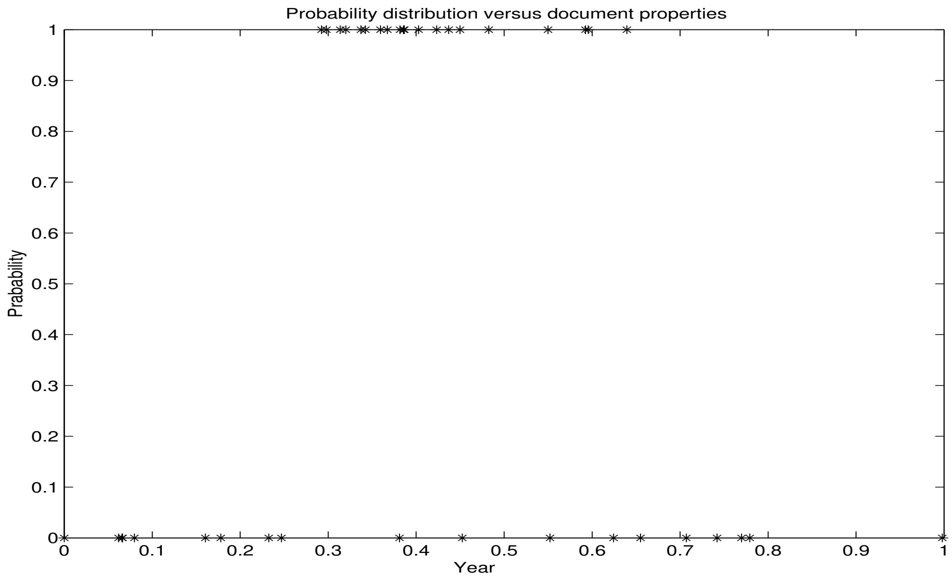



Figure 3.2: Scatter plot for the normalized term frequencies of index term **Year** in car-ads ontology using the training set documents

### 3.2.3 Random or Invariant Distribution

It is possible that data points are scattered randomly over the term frequency value domain, or the term frequencies are invariant (e.g., term frequencies are zero over all data points). In this case, it is impossible to conclude any functional relationship between the relevance probability and the corresponding index term. This type of data distribution is called a *random* or an *invariant* distribution. Figure 3.3 shows an artificially created scatter plot. Both $P = 1$ and $P = 0$ data points randomly distribute over the entire term frequency value domain (i.e., [0, 1]). Applying logistic regression on this data set will end up with a large $p$-value and small magnitude of regression coefficients, which indicates that the corresponding index term is insignificant. This type of index terms is not considered by our probabilistic retrieval model. In the car-ads application ontology, none of the index terms belong to this category. However, in the obituary application ontology, which we discuss in the next chapter, the lexical objects are more versatile, and term frequencies of several index terms are very small or approaching to zero regardless of whether a randomly chosen training document contains obituaries. The corresponding data distribution is similar to the one shown in Figure 3.3 or 3.4.

### 3.2.4 Complete Separation

In another extreme case, if an index term is a very strong indicator of the relevancy of a document, as shown in the scatter plot of the probability distribution with respect to term frequencies of the index term, the data points with $P = 1$ are completely separated from the data points with $P = 0$. In another words, there exists a vertical straight line that can totally separate these two groups of data points. This type of data distribution is called a *complete separation*. This situation is anticipated in the

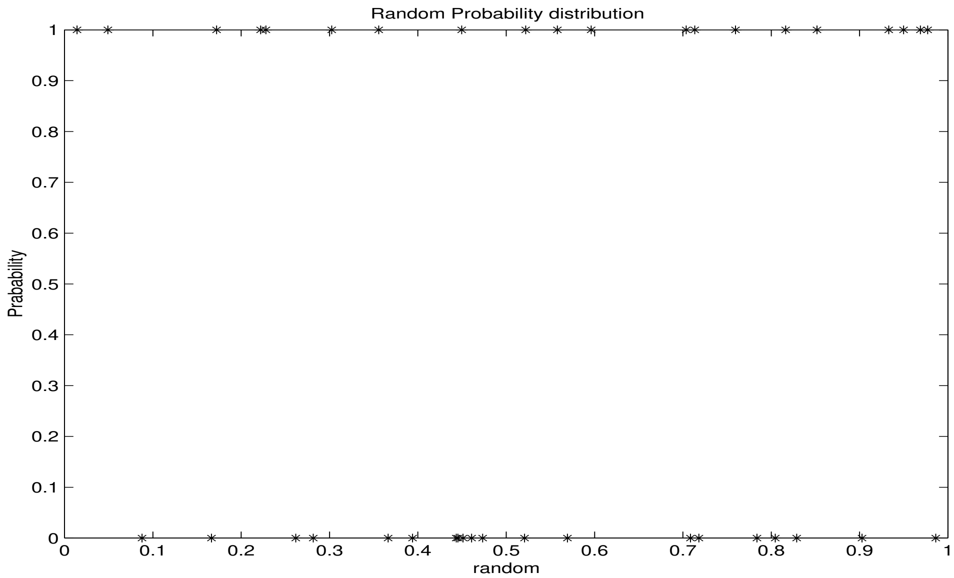



Figure 3.3: Scatter plot for an artificial random variable

Boolean model such that if the term frequency of a key index term (i.e., an important index term with respect to a user query) in a document is above a predefined threshold value, the corresponding document is categorized as relevant; otherwise, it is categorized as irrelevant with respect to that user query. This scenario, which is highly desirable in the Boolean model, is avoided in our probabilistic retrieval model using logistic regression. This is because in the pure sense of logistic regression, this complete separation causes logistic regression to diverge, i.e., the regression coefficients are either non-unique (i.e., it yields inconsistent values of regression coefficients for the same dependent and independent variables on the same input data file) or are unrealistically large. In addition, the $p$-value associated with the index term is close to one, which indicates the failure of logistic regression on this type of data distribution. The

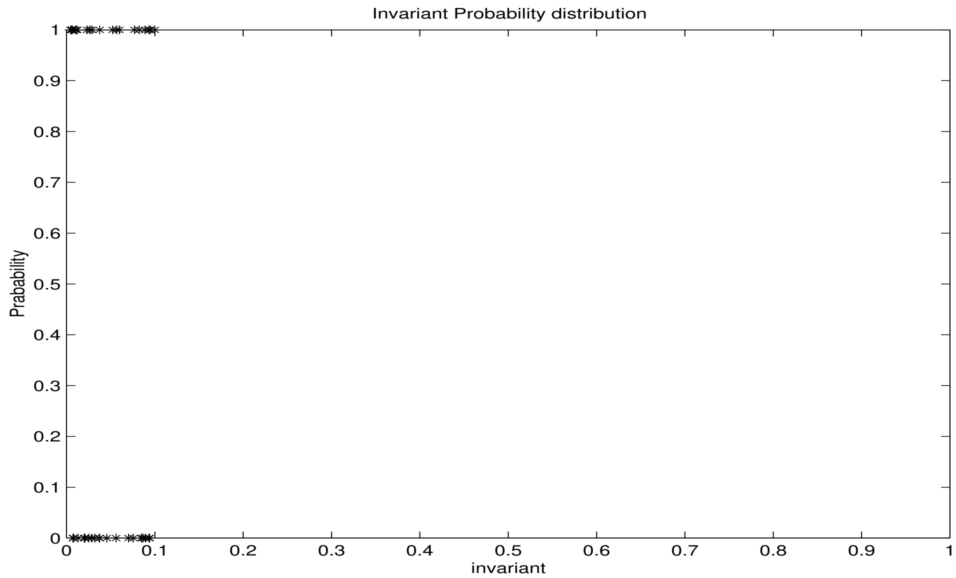



Figure 3.4: Scatter plot shown an invariant data distribution

problem of complete separation has been extensively studied before [BJ81, AA84]. The theoretical treatment of the problem is complicated and is purely mathematical and statistical in nature. The general idea behind the theoretical treatment is to show that under complete separation condition, the likelihood function does not have a maximum. As stated in Section 2.3.2, the regression coefficients are calculated by solving differential equations on regression coefficients. If the likelihood function does not have a maximum, there is no solution for these equations, and hence no valid regression coefficients.

From the statistical point of view, complete separation does not represent the real nature of the problem. As stated in [AA84], this type of problems is usually associated with *small* sample size. Theoretically, if we include a large number of documents in the training set, the complete separation problem could be resolved. However, in practice this problem is not easy to solve. First, there is no criterion on how large the sample size should be in order to avoid the problem. Second, we have no clue about what type (i.e., format or content) of documents tends to fill the gap between the separation. Fortunately, we have not encountered this problem in either the car-ads or obituary application ontology. This might imply that the sample sizes of the car-ads and obituary training document sets are appropriate.

Up till now, we have discussed all possible types of data distribution that we should consider in logistic regression analysis, and we are ready to apply logistic regression on the data listed in Table 3.1 to determine regression coefficients.

## 3.3 Logistic Regression on the Car-Ads Application Ontology

We evaluate the performance of our probabilistic model by calculating the precision, recall, and accuracy ratios using a collection of test documents. In order to verify the robustness of our model, we collect the training (and later the test) documents that are both "narrow" and "broad" in content. The content should be *broad* because we want our model to handle different kinds of documents. The content should be *narrow* because we would like to catch subtle differences between documents.

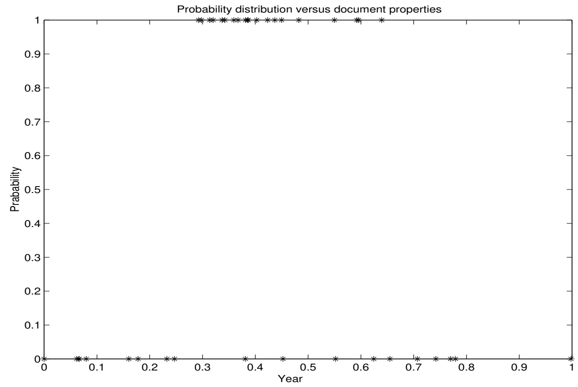
### 3.3.1 Regression Coefficients of the Index Terms, the Density Value, and the Grouping Value of Car-Ads Training Documents

Figures 3.5 to 3.13 show the scatter plots of the term frequencies of different index terms, the density value, and the grouping value derived from the training set documents according to the car-ads application ontology. From these figures, we observe that none of the index terms has a random, invariant, or complete separation distribution, which means that all the corresponding lexical objects defined by their corresponding lexical object sets in the car-ads application ontology are statistically important in categorizing car-ads documents. Note that the index term **Year** has a clear sign of double $S$-curve distribution, and hence its regression analysis are performed on two different value domains (i.e., $< 0.39$ and $\geq 0.39$). All the other plots show a clear single $S$-curve distribution with a small overlap. Applying logistic regression on these index terms should yield valid regression coefficients.

The logistic regression package that has been used for computing regression coefficients in our probabilistic retrieval model is included in **SPSS 10.0** for Windows. Logistic regression function is installed under the Pull-Down menu **Analyze** and within its submenus **Regression**, **Binary logistic ...**. The format of the input data file corresponds to Table 3.1, with values separated by a space. Figure 3.14 shows the Pull-Down menu, as well as the input values in a Spreadsheet format, whereas Figure 3.15 displays the dialogue window for logistic regression, where logistic regression is applied to the index term **Year**.

For illustration, we select the **Probability** column as the dependent variable and the **Year** column as the independent variable in logistic regression. Figure 3.16 shows the results of applying logistic regression on **Year**. The regression coefficients of **Year** are listed in the second column (i.e., B) of the bottom table where $C_1 = .572$ and $C_0 = -.230$. The corresponding $p$-value (i.e., .679) is listed in the sixth column (i.e., Sig.) of the bottom table. The regression coefficients and the corresponding $p$-values for other quantities shown in Figure 3.14 can be computed accordingly. Table 3.3.1 lists the regression coefficients and the associated $p$-values for all index terms, along with the density value $y$ and the grouping value $z$, in the car-ads ontology. The last row in Table 3.3.1 lists the term frequencies, density value, and grouping value for the first document in the test document set, which is partially shown in Figure 3.17.

In Table 3.3.1, a single logistic regression is applied to all document properties (i.e., the index terms along with the density and grouping values). From the data in the table, we can study the role of each index term in the determination of the relevance probability of a document with respect to the application ontology. If the $p$-value of an index term is much larger than 5%, the index term is subject to further investigation. If the large $p$-value is caused by random or invariant distribution, the corresponding index term should be excluded. However, if it is caused by the double $S$-curve distribution, the regression analysis should be carried out in the two term frequency value domains (e.g., $< 0.39$ and $\geq 0.39$ for the index term **Year**).



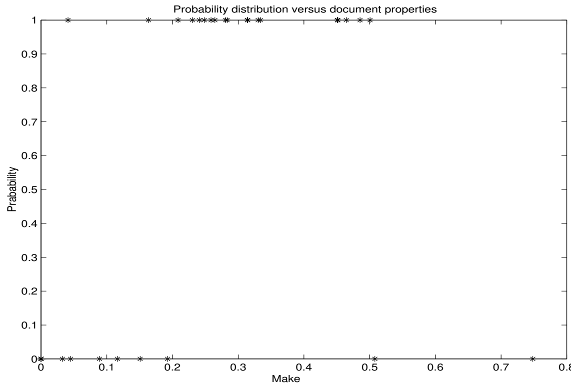
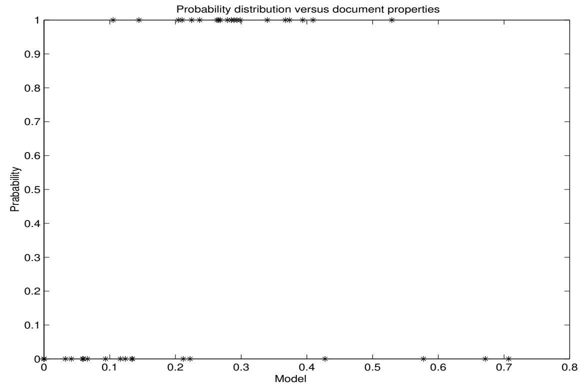




Figure 3.5: Scatter plot for the term frequencies of the index term **Year**

Figure 3.6: Scatter plot for the term frequencies of the index term **Make**



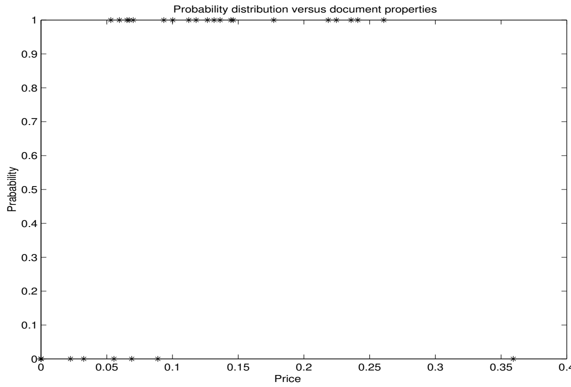
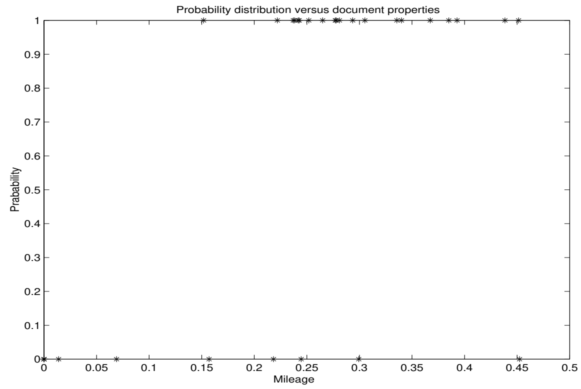




Figure 3.7: Scatter plot for the term frequencies of the index term **Model**

Figure 3.8: Scatter plot for the term frequencies of the index term **Mileage**



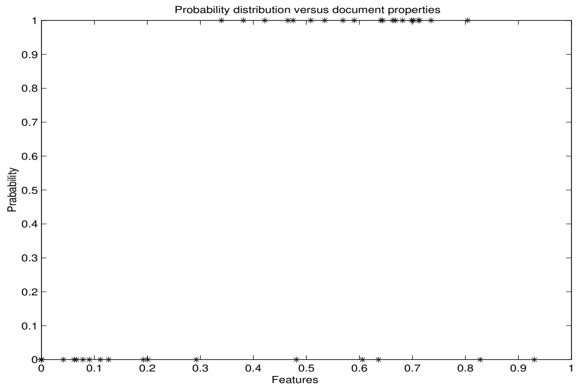




Figure 3.9: Scatter plot for the term frequencies of the index term **Price**

Figure 3.10: Scatter plot for the term frequencies of the index term **Feature**

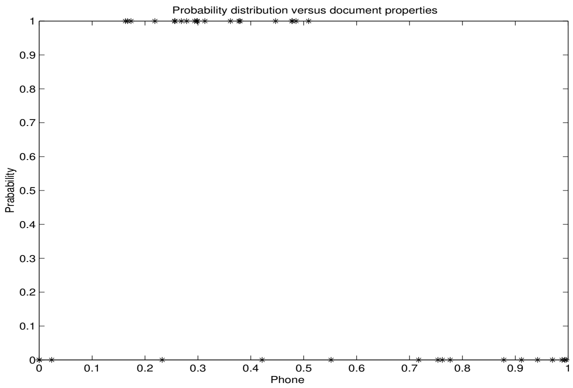



Figure 3.11: Scatter plot for the term frequencies of the index term **Phone**

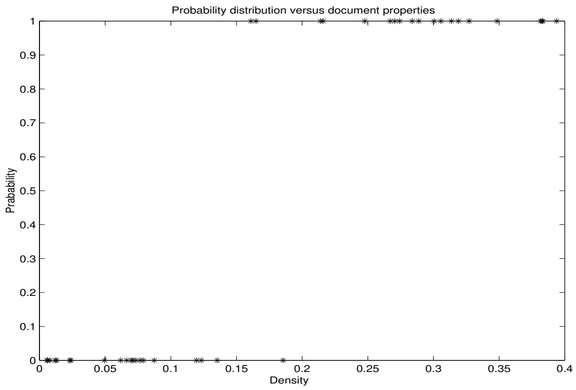



Figure 3.12: Scatter plot for the **density** heuristic value $y$

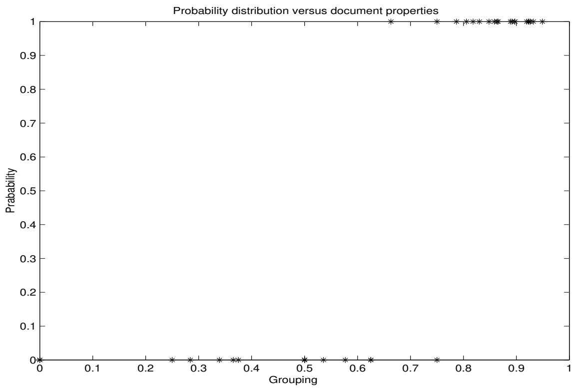



Figure 3.13: Scatter plot for the **grouping** heuristic value $z$

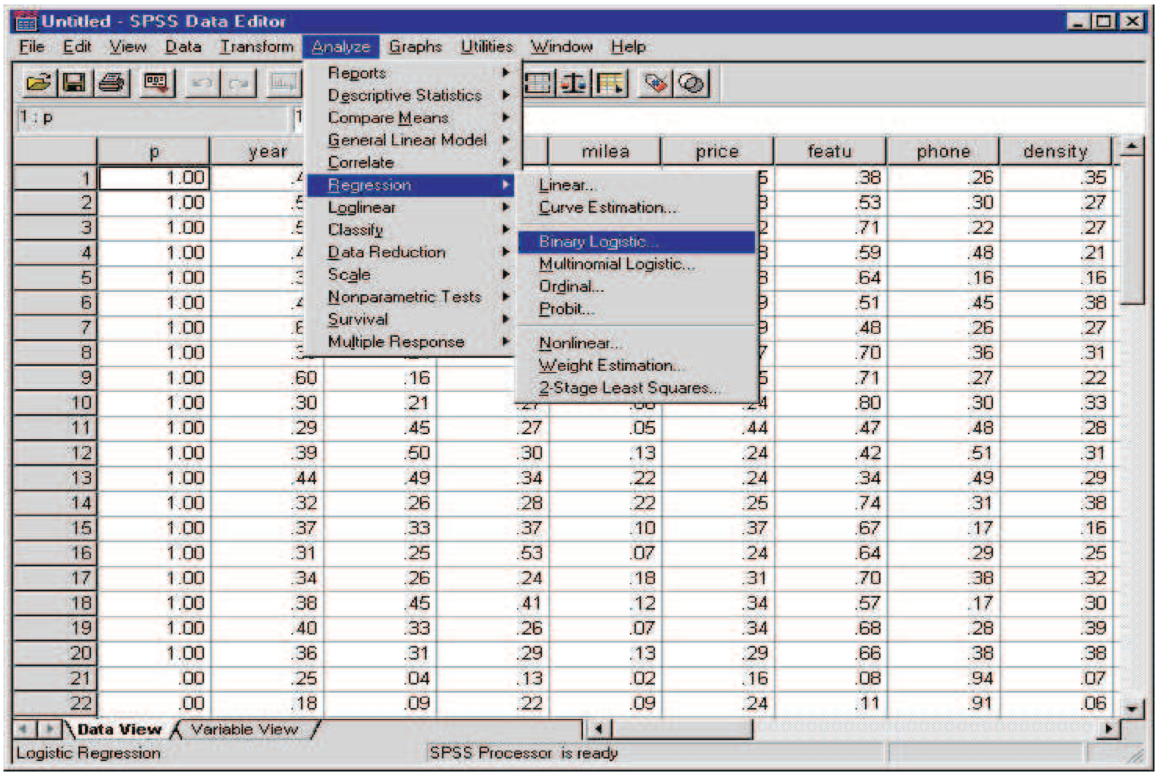



Figure 3.14: Pull-down menu for logistic regression on an input data file for the car-ads ontology in SPSS

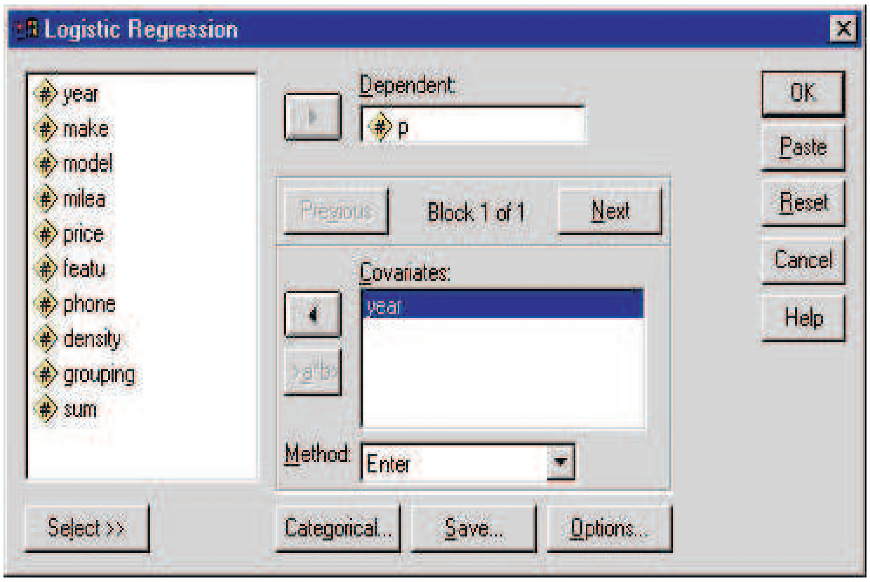



Figure 3.15: Dialogue box for logistic regression on the index term **Year** in SPSS

In the car-ads application ontology, using the single $S$-curve-fitting, all $p$-values of the index terms in the training document set, except **Year**, are below or close to 5%. The $p$-values indicate that all the lexical object (i.e., index terms) of the car-ads ontology, except **Year**, are important in dichotomizing positive and negative documents with respect to the car-ads ontology. The $p$-value for the index term **Year** is 0.68, which is much larger than the acceptable risk factor 0.05. However, this does

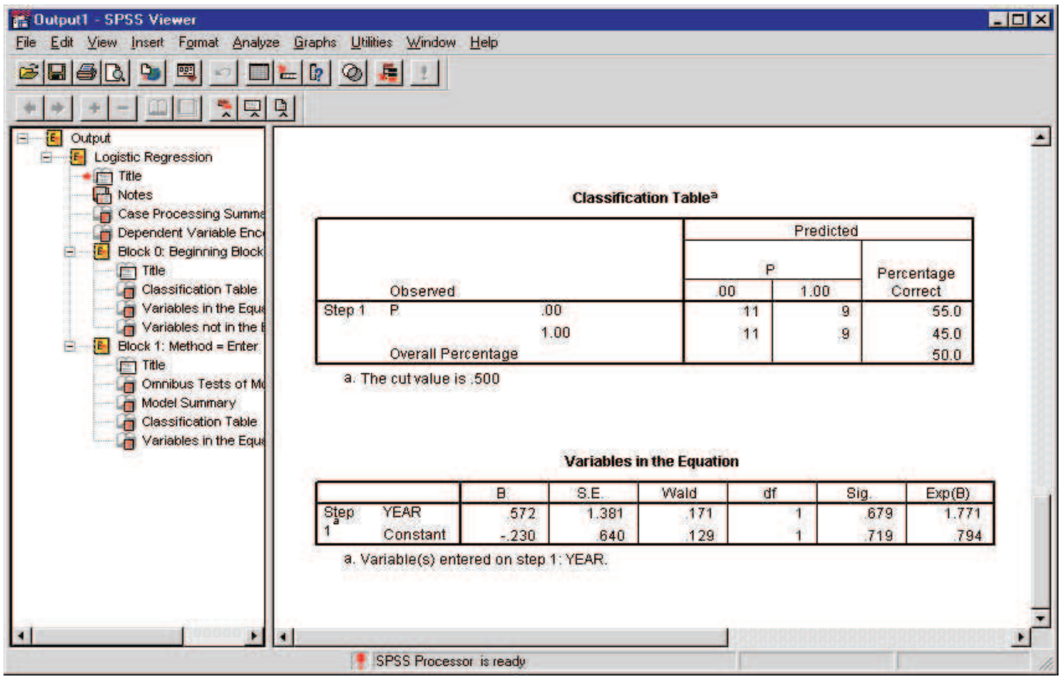



Figure 3.16: Results generated by logistic regression on the index term **Year** in SPSS

not mean that **Year** is not an important index term for categorizing the relevance of a document with respect to the car-ads ontology. The problem of this large $p$-value is caused by the fact that we have used a single $S$-curve to fit a double $S$-curve distribution whose pattern is clearly shown in the scatter plot of **Year** in Figure 3.5.

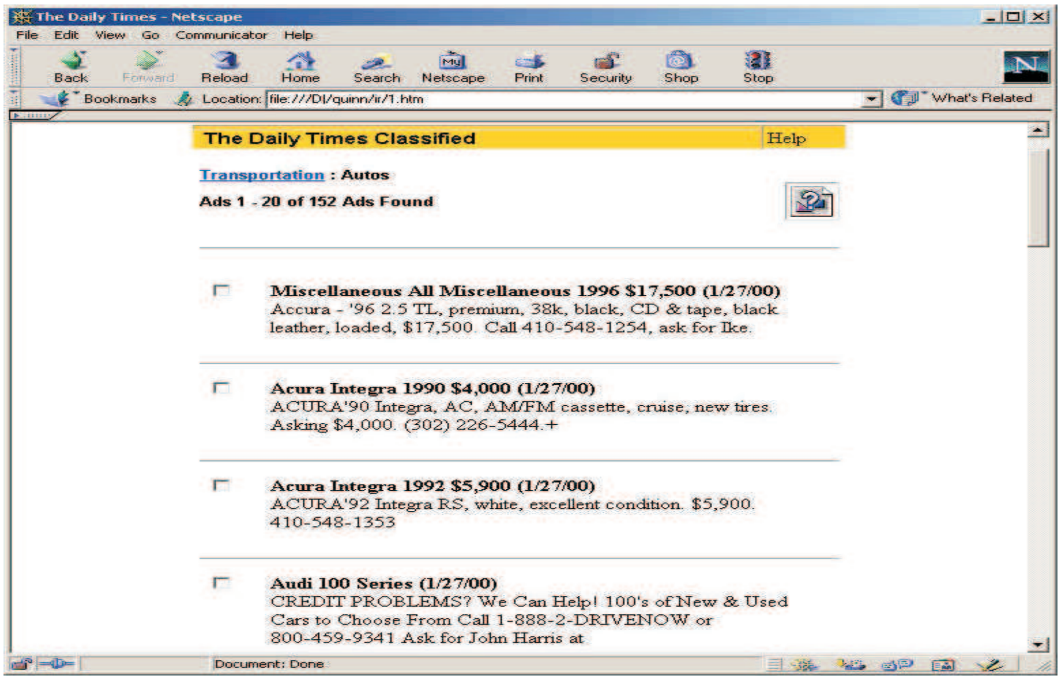



Figure 3.17: Car advertisements retrieved from http://www.delmarvaclassfieds.com

In order to fit a double $S$-curve distribution of an index term, such as **Year**, we first sort the training documents according to their term frequency values of the index term. The term frequency value of 0.39 splits the training document set in half. After we have split the set of training documents into two disjoint subsets, we perform two separate logistic regression analyses on the index term. Consider the index term *Year*. As shown in Table 3.3, the $p$-values of the index term **Year** in the two different value domains all fall well below 5%. With that, we claim that we have computed all the regression coefficients of each "significant" index term, the density value, and the grouping value in the car-ads ontology for our probabilistic retrieval model.

In Equation 2.24, another quantity that should be computed is $\ln(O(R))$, which is the logodds of the probability that a randomly chosen document in a training document set is relevant to the corresponding application ontology. The car-ads training set consists of twenty positive and twenty negative documents. Thus, the probability that a randomly chosen document in the training set is relevant to the car-ads application ontology is 50%. Using the definitions of odds and logodds of an event, we compute

$$\begin{aligned} P(r) &= 0.5 \\ O(R) &= \frac{P(R)}{P(\bar{R})} = \frac{P(R)}{1 - P(R)} = \frac{0.5}{1 - 0.5} = 1 \\ \ln(O(R)) &= \ln(1) = 0. \end{aligned}$$

Thus, we substitute $\ln(O(R))$ by zero in Equation 2.24. Using the regression coefficients listed in Tables 3.3.1 and 3.3, we can estimate the relevance probability of a document with respect to the car-ads ontology.

### 3.3.2 Probabilistic Retrieval of Car-ads and Non-car-ads Documents Based on Logistic Regression Analysis

We have been using the car-ads application ontology as an example to illustrate how the proposed probabilistic retrieval model works. Given an arbitrary Web document $d$, we determine whether $d$ is a car-ad by estimating $P(R|d)$, which is the relevance probability of $d$ with respect to the car-ads ontology. As shown in Table 3.3.1, the regression coefficients of the index term **Make** are $C_0 = -1.61$ and $C_1 = 8.36$. For the first document in the car-ads test set, the term-frequency of **Make** is 0.46, i.e., $x_2 = 0.46$. By substituting the values of $x_2$, $C_0$, and $C_1$ into Equation 2.26, it yields

$$\begin{aligned} \ln(O(R|Make)) &= C_0 + C_1 \times x_2 \\ &= -1.61 + 8.36 \times 0.46 \\ &= 2.24. \end{aligned}$$

Since $\ln(O(R|Make)) > 0$, the term frequency value of **Make** favors the result that the first test document is relevant to the car-ads ontology. However, the document relevancy is jointly determined by the other index terms along with the density value $y$ and grouping value $z$. Thus, the logodds of other index terms along with $y$ and $z$ should be calculated accordingly.

In writing a program to compute relevance probabilities, it is convenient to express regression coefficients and term frequencies of index terms of a document as vectors. According to the data in Tables 3.1 and 3.3.1, we construct three row vectors for any document in the test set as follows:

$$\begin{aligned} C_1 &= (C_{11}, \cdots, C_{1n}, C_{y1}, C_{z1}) \\ C_0 &= (C_{01}, \cdots, C_{0n}, C_{y0}, C_{z0}) \\ X &= (x_1, \cdots, x_n, y, z). \end{aligned}$$

The length of each vector is $n + 2$, where $n$ is the number of index terms and the two additional components are the density and grouping values. The components of $C_0$, $C_1$, and $X$ are the first, second, and fourth row of Table 3.3.1, respectively. The summation in Equation 2.24 can be expressed as

$$\begin{aligned}
\sum_{i=1}^{n}[\ln(O(R|x_i))]] + \ln(O(R|y)) + \ln(O(R|z)) &= \sum_{i=1}^{n+2}(C_{0i} + C_{1i} \cdot X_i) \qquad (3.1)\\
&= \sum_{i=1}^{n+2} C_{0i} + \sum_{1}^{n} C_{1i} \cdot X_i\\
&= \sum_{i=1}^{n+2} C_{0i} + C_1 \cdot X^T .
\end{aligned}$$

Substituting this summation in Equation 2.24 yields

$$\ln(O(R|x_1, \cdots, x_n, y, z) = 25.66 \qquad (3.2)$$

Using the definition of logodds i.e., Equation 2.29, we obtain

$$\begin{aligned}
P(R|x_1, \cdots, x_n, y, z)) &= \frac{1}{1 + \exp(-25.66)}\\
&\simeq 1
\end{aligned}$$

which is the estimated relevance probability of the first test document $d$ whose term frequencies of various index terms are listed in Table 3.3.1 with respect to the car-ads application ontology. Since $p \simeq 1$, we classify $d$ as relevant, which turns out to be the correct answer.

## 3.4 Relevance Probability Estimates of the Car-ads Test Set Documents

The relevance probability of test documents are calculated using the method described in Section 3.3.2. For each test document, the corresponding document vector $X$ is substituted into Equations 3.1. Then the relevance probability is calculated using Equations 2.24 and the definition of logodds. Using the car-ads application ontology as an example, the calculated relevance probabilities of the twenty test documents are shown in Tables 3.4 and 3.4.

Using a probabilistic retrieval model, the natural threshold value to discern relevant from irrelevant documents is set to 50%. A perfect relevant or irrelevant document is the one with near asymptotic probability value (i.e., zero for negative documents and one for positive documents). In the car-ads application ontology, our model misclassifies only one document in the test set. The tenth document in the negative test set is an advertisement for antique cars (see the last row in Table 3.4). However, our probabilistic retrieval model classifies it as a car-ads document. By carefully scrutinizing the content of the antique car document, we notice that it has the multiple-record structure just like car-ads documents. Within the document, most records contain manufacture year, make, model, price, mileage, and features, etc. The corresponding density and grouping values are close to the values that appear in a real car-ads document. All these features of the antique care document match the car-ads application ontology. Hence, our probabilistic retrieval model cannot distinguish one from another. The eighth document (i.e., the third row from the bottom of Table 3.4) in the negative test set is an advertisement of motorcycles. Even though the document format and content are also very close to a car-ads document, the domain values of **Make** and **Model** are different from those defined in the car-ads application ontology. Our model is able to detect the difference and categorize it as an irrelevant document with respect to car-ads since its relevance probability is barely 10%. Further consider the sixth document in the negative test set, which is an advertisement of Items for Sale or Rent. It is a multiple-record document. Some records in the document contain phone number and some item descriptions are extracted as car features. Our model categorizes it correctly with the relevance probability of 14%. For the remaining test documents, our model can categorize them correctly with near asymptotic probability values.

## 3.5 Model Refinement by Including Correlation Relations in Relevance Probability Calculation

Even though our probabilistic retrieval model has a high accuracy on the test set of the car-ads application ontology, there is still room to improve. For example, the relevance probability of the sixth and eighth documents in the negative test set do not converge to zero, even though they are not car-ads documents. Up till now, we have been treating all index terms in the car-ads application ontology as uncorrelated quantities. The relevance probability of a document with respect to the ontology is determined by the accumulated, but independent, contributions from the index terms as well as the density and grouping values. We realize that there might be some implicit information buried inside the data as listed in Table 3.1 that could potentially improve the performance of our model. When we scrutinize Table 3.1 again, we notice that every index term follows a similar pattern: the term frequency values are usually *large* for positive documents and *small* for negative ones. This suggests the existence of strong positive correlations[2] among the index terms in the car-ads ontology. The strong correlation relations were further verified by the correlation coefficients calculated by the correlation function provided by **SPSS 10.0**. This discovery inspires us to include the correlation relations among index terms in the car-ads ontology. Based on the data listed in Table 3.1, we construct twenty-one extra columns that consist of any possible product of two term frequencies of the index terms. (There are seven index terms defined in the car-ads ontology, and hence the number of unique combinations is $C_7^2 = 21$.) We apply logistic regression analysis on these twenty-one data columns the same way as we apply to the term frequency columns of the index terms. The corresponding regression coefficients and $p$-values are shown in Tables 3.6. The $p$-values indicate that fourteen (i.e., **Year-Make, Year-Mileage, Year-Price, Year-Feature, Make-Model, Make-Price, Make-Feature, Make-Phone, Model-Mileage, Model-Price, Model-Feature, Mileage-Price, Mileage-Feature, Feature-Phone**) out of the twenty-one possible combinations are significant. Following the procedure illustrated in Section 3.3.2, we construct three vectors of $C_0$, $C_1$, and $X$ with the expanded size. The new vector size is equal to $n$ (the number of chosen index terms) + 2 (the density

[2]Correlation is measured by the correlation coefficient $r$ such that $r = 1$ indicates strong positive correlation, $r = -1$ indicates strong negative correlation, and $r = 0$ indicates no correlation.

and grouping values) + $Sig(C_n^2)$, which is the number of the significant combinations of index terms. For the car-ads ontology, the new vector size is $7 + 2 + 14 = 23$. Tables 3.5 and 3.5 show the results when the important correlation terms are considered.

As shown in Tables 3.5 and 3.5, the relevance probability of the car-ads test documents is one for each positive test document and closer to zero for each negative document (see the probability of the sixth and eighth documents, i.e., the Items for Sale or Rent and the motorcycles documents, in Table 3.5). The results indicate that we have improved the performance of our model on the car-ads application ontology.

| Probability | Year | Make | Model | Mile | Price | Feature | Phone | y | z |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.48 | 0.45 | 0.37 | 0.09 | 0.45 | 0.38 | 0.26 | 0.35 | 0.92 |
| 1 | 0.59 | 0.32 | 0.21 | 0.24 | 0.28 | 0.54 | 0.30 | 0.27 | 0.82 |
| 1 | 0.55 | 0.23 | 0.15 | 0.14 | 0.22 | 0.71 | 0.22 | 0.27 | 0.79 |
| 1 | 0.45 | 0.28 | 0.23 | 0.11 | 0.28 | 0.59 | 0.48 | 0.21 | 0.75 |
| 1 | 0.34 | 0.47 | 0.29 | 0.26 | 0.28 | 0.64 | 0.16 | 0.16 | 0.83 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 0 | 0.24 | 0.05 | 0.14 | 0.02 | 0.16 | 0.08 | 0.94 | 0.07 | 0.58 |
| 0 | 0.18 | 0.09 | 0.22 | 0.09 | 0.25 | 0.11 | 0.91 | 0.06 | 0.63 |
| 0 | 0.08 | 0.00 | 0.06 | 0.36 | 0.30 | 0.00 | 0.88 | 0.08 | 0.63 |
| 0 | 0.45 | 0.15 | 0.00 | 0.00 | 0.45 | 0.00 | 0.75 | 0.19 | 0.50 |
| 0 | 0.06 | 0.00 | 0.12 | 0.00 | 0.00 | 0.06 | 0.99 | 0.07 | 0.00 |
| 0 | 0.07 | 0.00 | 0.67 | 0.00 | 0.00 | 0.61 | 0.42 | 0.05 | 0.34 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |

Table 3.1: Normalized term frequencies along with the density value ($y$) and grouping value ($z$) of the training set documents according to the car-ads ontology

| | Year | Make | Model | Mileage | Price | Feature | Phone | y | z |
|---|---|---|---|---|---|---|---|---|---|
| $C_0$ | -.23 | -1.61 | -.86 | -1.66 | -3.01 | -2.55 | 1.06 | -10.13 | -20.45 |
| $C_1$ | .57 | 8.36 | 3.70 | 22.82 | 15.52 | 5.93 | -2.46 | 61.91 | 29.22 |
| $p$-value | .68 | 0.00 | 0.07 | 0.00 | 0.00 | 0.00 | 0.03 | 0.05 | 0.01 |
| Term frequency | 0.48 | 0.46 | 0.26 | 0.42 | 0.12 | 0.48 | 0.27 | 0.33 | 0.89 |

Table 3.2: Regression coefficients ($C_0$ and $C_1$), $p$-values of index terms, the density (y) and grouping (z) values of the car-ads ontology, and the term frequencies of the first test document for the car-ads ontology (Table B.1 in Appendix B contains the data of all the car-ads test set documents)

| | $t - f < 0.39$ | $t - f \geq 0.39$ |
|---|---|---|
| $C_0$ | -7.711 | 8.949 |
| $C_1$ | 28.179 | -15.093 |
| $P$-value | 0.039 | 0.019 |

Table 3.3: Regression coefficients of the index term **Year** computed by using double $S$-curve distribution

| Document Region | URL | Probability |
|---|---|---|
| East Coast | http://www.delmarvaclassfieds.com | 1.00 |
| North East Central | http://www.thetelegraph.com | 1.00 |
| North East Coast | http://www.vermontclassifieds.com | 0.99 |
| North West Central | http://www.ndweb.com/mdnonline/mdnonline.html | 1.00 |
| Pacific North West | http://www.adn.com | 1.00 |
| Pacific South West | http://www.hawaiisnews.com/cars | 1.00 |
| South East Central | http://www.brewtonstandard.com | 1.00 |
| South East Coast | http://www.aikenstandard.com/ | 1.00 |
| South West Central | http://adaeveningnews.com | 0.99 |
| West Coast | http://www.tahoe.com/classifieds/tdt/9100.html | 1 |

Table 3.4: Positive test documents and their relevant probabilities of the car-ads ontology

| Document Subject | URL | Probability |
|---|---|---|
| Person | http://www.cs.byu.edu/infor/dwembly.html | 2.0e-08 |
| Search Engine | http://www.dogpile.com | 1.3e-05 |
| Purchase Form | http://www.ecampus.com | 5.9e-11 |
| Missing People | http://www.cyberpages.com | 1.3e-05 |
| Jobs | http://www.ohio.com | 1.7e-09 |
| Items for Sale or Rent | http://www.crookstontimes.com | 0.14246 |
| Personals | http://www.date-net.com/index.CFM | 1.1e-07 |
| Motorcycles | http://www.netbikes.yks.com/general.html | 0.10227 |
| Boats | http://www.sunspot.net/ | 0.00007 |
| Antique Cars | http://www.internetclassiccars.com/ files/twenties0.html | 0.90317 |

Table 3.5: Negative test documents and their relevant probabilities of the car-ads ontology

| | Year-Make | Year-Model | Year-Mileage | Year-Price | Year-Feature | Year-Phone | Make-Model |
|---|---|---|---|---|---|---|---|
| $C_0$ | 27.15 | 4.51 | 117.70 | 36.62 | 11.45 | -.12 | 21.55 |
| $C_1$ | -2.06 | -.42 | -3.09 | -2.57 | -1.87 | .02 | -1.09 |
| $p$-value | .00 | 0.24 | 0.00 | 0.00 | 0.00 | 0.96 | 0.01 |
| | Make-Mileage | Make-Price | Make-Feature | Make-Phone | Model-Mileage | Model-Price | Model-Feature |
| $C_0$ | 13080.84 | 84.33 | 23.60 | 16.67 | 223.56 | 99.50 | 17.06 |
| $C_1$ | -116.49 | -2.87 | -2.34 | -.98 | -3.17 | -3.55 | -1.79 |
| $p$-value | .98 | 0.00 | 0.00 | 0.02 | 0.00 | 0.00 | 0.00 |
| | Model-Mileage | Mileage-Price | Mileage-Feature | Mileage-Phone | Price-Feature | Price-Phone | Feature-Phone |
| $C_0$ | 9.88 | 79.88 | 120.81 | 9.98 | 203.51 | 6.58 | 20.17 |
| $C_1$ | -.73 | -1.57 | -3.61 | -.32 | -84.00 | -.50 | -2.61 |
| $p$-value | .08 | 0.00 | 0.00 | 0.26 | 0.99 | 0.14 | .00 |

Table 3.6: Regression coefficients ($C_0$ and $C_1$) and $p$-values of correlation relations in the car-ads ontology

| Document Region | URL | Probability |
|---|---|---|
| East Coast | http://www.delmarvaclassfieds.com | 1.00 |
| North East Central | http://www.thetelegraph.com | 1.00 |
| North East Coast | http://www.vermontclassifieds.com | 1.00 |
| North West Central | http://www.ndweb.com/mdnonline/mdnonline.html | 1.00 |
| Pacific North West | http://www.adn.com | 1.00 |
| Pacific South West | http://www.hawaiisnews.com/cars | 1.00 |
| South East Central | http://www.brewtonstandard.com | 1.00 |
| South East Coast | http://www.aikenstandard.com/ | 1.00 |
| South West Central | http://adaeveningnews.com | 1.00 |
| West Coast | http://www.tahoe.com/classifieds/tdt/9100.html | 1.00 |

Table 3.7: Relevance probabilities of positive test documents, including the correlations among the index terms, of the car-ads ontology

| Document Subject | URL | Probability |
|---|---|---|
| Person | http://www.cs.byu.edu/infor/dwembly.html | 2.0e-22 |
| Search Engine | http://www.dogpile.com | 1.2e-28 |
| Purchase Form | http://www.ecampus.com | 6.3e-29 |
| Missing People | http://www.cyberpages.com | 5.5e-19 |
| Jobs | http://www.ohio.com | 2.0e-21 |
| Items for Sale or Rent | http://www.crookstontimes.com | 4.1e-11 |
| Personals | http://www.date-net.com/index.CFM | 3.6e-22 |
| Motorcycles | http://www.netbikes.yks.com/general.html | 0.00519 |
| Boats | http://www.sunspot.net/ | 5.1e-11 |
| Antique Cars | http://www.internetclassiccars.com/files /twenties0.html | 1 |

Table 3.8: Relevance probabilities of negative test documents, including the correlations among the index terms, of the car-ads ontology

# Chapter 4

# Experimental Results

In this chapter, we evaluate the performance of our proposed probabilistic retrieval model using the experimental results on documents in a test set. The model was tested on two different application ontologies, car-ads and obituary. Both the car-ads and obituary test sets contain ten positive and ten negative documents. The model performance is evaluated by computing the recall, precision, and accuracy ratios on each test document set.

From the experimental results, we notice that some irrelevant documents with respect to an application ontology in the test document set have very similar document structure (i.e., multiple-record) and content as the relevant ones. These documents tend to cause false positives[1]. For example, advertisements of motorcycle, boat,

[1] There are two different kinds of false categorizations in IR. One is *false positive* where an irrelevant document is categorized as relevant. The other one is *false drop* where a relevant document is categorized as irrelevant.

and snowmobile for-sale have a similar ontology structure as the car-ads application ontology. As a result, data extracted from these documents are similar to the data anticipated in a car-ads document, and our probabilistic retrieval model might fail to distinguish these non-car-ads documents from real car-ads documents. Based on this observation, we purposely selected some similar Web documents that might cause false positives. We have analyzed the experimental results and provided explanations for the performance of our model on these similar documents.

## 4.1 Experimental Results on the Car-Ads Application Ontology

The process of computing the relevance probability of test documents with respect to the car-ads application ontology has been discussed in Chapter 3, and the results of applying our probabilistic retrieval model to distinguish relevant and irrelevant test documents are listed in Tables 3.4 and 3.4. Since the ten positive documents are all correctly categorized and there is only one false positive from the ten negative documents, the corresponding recall ratio is $\frac{10}{10} = 100\%$, the precision ratio is $\frac{10}{10+1} = 90.9\%$, and the accuracy ratio is $\frac{20-1}{20} = 95\%$. The false positive is an antique car Web document. As mentioned earlier, since the document structure and content of the antique car Web document are very similar to a car-ads document, our model treats it as a car-ad document.

| Document | URL | Probability |
|---|---|---|
| Arctic | http://www.photoads.com/sleds/usedsled/arctic/arctic.htm | 0.00 |
| Bicycle | http://www.cyberus.ca/~obcweb/ | 0.00 |
| Boat-snowmobile | http://www.prairietech.net/midweek_classified_ads.htm#110-110 BOATS | 0.84 |
| Polaris | http://www.photoads.com/sleds/usedsled/polaris/polaris.htm | 0.02 |
| PWC | http://www.photoads.com/boats/usedboat/pwc/pwc.htm | 0.00 |
| Sailboat | http://www.photoads.com/boats/usedboat/sailboat/sailboat.htm | 0.01 |
| Sale-rent | http://www.crookstontimes.com/classifieds/ | 0.00 |
| Special MC | http://www.photoads.com/cycles/usedmc/usedatv/special.htm | 0.05 |

Table 4.1: Similar test documents for the car-ads application ontology

In order to further investigate the performance of our model on documents similar to car-ads documents such as motorcycle-for-sale, boat-for-sale, and snowmobile-for-sale, etc., we purposely selected eight other similar documents. Our probabilistic model correctly categorizes seven out of these eight documents. Table 4.1 shows the relevance probabilities of these eight documents computed by our model. Note that the third document in Table 4.1 is a false positive, which is a boat-snowmobile-for-sale document.

With the inclusion of the eight similar documents, the mis-classification made by our probabilistic model includes two false positives (i.e., the antique car and boat-snowmobile Web documents). The recall ratio of our model on the car-ads application remains 100%, the precision ratio drops to $\frac{10}{10+2} = 83.3\%$, and the accuracy ratio reduces to $\frac{28-2}{28} = 92.9\%$. The performance of our model could be better than these numbers might imply in general since more than half of the negative test documents (i.e., 10 out of 18) are very similar documents that tend to cause false positives.

## 4.2 Experimental Results on the Obituary Application Ontology

Figure 4.1 shows the graphical representation of the obituary ontology [ENX01]. As shown in the figure, the obituary ontology structure is more complicated than the car-ads ontology. When we examined the forty documents in the obituary training set, we noticed that obituary documents have a more versatile format than the car-ads training set documents, which is expected. For instance, some obituary documents include **Relative Name** and **Relationship** with the deceased person, whereas others do not. Some obituary documents contain **Birth Date** and **Death Date** without **Age** (even though age is implicitly stated in this case). Furthermore, many of these documents do not include **Viewing Address** and **Viewing Date** at all. These instances indicate that some lexical objects in obituary correspond to either the random or invariant data distribution as described in Section 3.2.

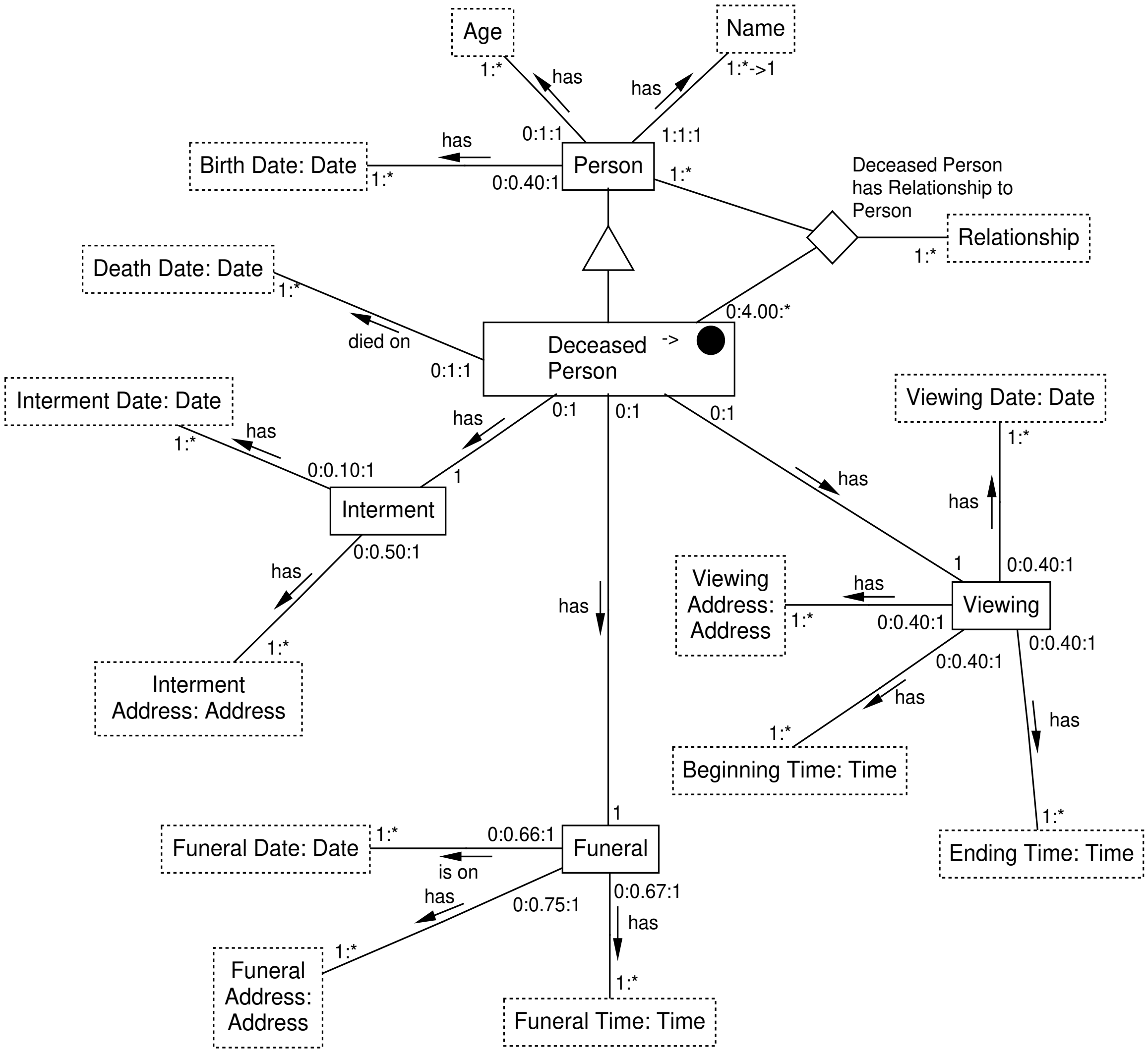



Figure 4.1: Graphical representation of the obituary application ontology

The training data of the obituary application ontology was extracted from a training document set with twenty positive and twenty negative obituary documents. Contrary to the car-ads application ontology, where all lexical objects in the ontology carry more or less the same weight in categorizing documents, more than one-third of lexical objects in the obituary ontology do not have any statistical significance in term of logistic regression analysis on the obituary training document set. This problem is caused by the fact that obituary documents are versatile. Many obituary documents in the training and test sets contain only a subset of the lexical objects defined in the obituary ontology. We have tested the performance of our probabilistic retrieval model using the obituary ontology on the training and test set documents. The extracted data shows that the term frequencies of **Viewing Date, Viewing Address, Funeral Date, Funeral Time**, and **Interment Date** are virtually zero in the entire training document set. Logistic regression analysis on these lexical objects yields near zero regression coefficients, which means that based on our obituary training documents they are less significant than other lexical objects of the obituary ontology in categorizing obituary documents.

After the relatively insignificant lexical objects were excluded, the training data used for the logistic regression analysis on the obituary ontology was normalized using the method as described in Chapter 3. Table 4.2 shows the regression coefficients $C_0$ and $C_1$, and the corresponding $p$-values for the obituary training documents. As shown in the table, the $p$-values for **Deceased Name** and **Ending Time** are 0.47 and 0.55, respectively. Figures 4.2 and 4.3 show their corresponding scatter plots. The data distribution of the **Ending Time** is neither a single nor double $S$-curve distribution. Instead, it is closer to random distribution, and hence was excluded from the list of chosen index terms. The double $S$-curve fit was tried on **Deceased**

| | Deceased Name | Age | Death Date | Birth Date | Beginn- ing Time |
|---|---|---|---|---|---|
| $C_1$ | 0.9 | 8.8 | 36.4 | 1.6 | 2.7 |
| $C_0$ | -0.4 | -2.3 | -3.0 | -0.4 | -0.6 |
| $p$-value | .47 | .00 | .00 | .17 | .05 |
| | Ending Time | Funeral Address | Interment Address | Den- sity | Group- ing |
| $C_1$ | -1.4 | 8.1 | -7.2 | 11.9 | 20.3 |
| $C_0$ | 0.1 | -0.3 | 2.6 | -3.2 | -13.5 |
| $p$-value | .55 | .15 | .00 | .00 | .00 |

Table 4.2: Initial regression coefficients and *p-values* of significant index terms in the obituary application ontology using the obituary training documents

**Name**. By splitting the data points in half using the value of 0.31, we applied logistic regression on the two term frequency value domains (i.e., [0, 0.31) and [0.31, 1]), which yields $C_0$, $C_1$, and $p$-values as shown in Table 4.3. Even though the $p$-value in the interval [0.31, 1] is 0.33, the $p$-value in the other interval is significantly reduced, which means that the double $S$-curve fit is more appropriate than the singe $S$-curve.

The relevance probabilities of all the obituary test documents are listed in Tables 4.2 and 4.2. The recall, precision, and accuracy ratios are 100%, 90.9%, and 95%, respectively.

Table 4.6 shows the comparison of the number of occurrences of the index terms

| | $t - f < 0.31$ | $t - f \geq 0.31$ |
|---|---|---|
| $C_0$ | -2.052 | 2.326 |
| $C_1$ | 9.295 | -3.333 |
| $P$-value | 0.077 | 0.333 |

Table 4.3: Regression coefficients of index term **Deceased Name** using a double $S$-curve distribution

along with the density and grouping values extracted from an obituary document and the motorcycle-for-sale Web document. Our model classifies these two documents to be in the same document category. Because the product of the term frequencies of **Death Date** and **Birth Date** is large in the motorcycle-for-sale document, the correlation between **Death Date** and **Birth Date** does not contribute in discerning the motorcycle document from obituaries. However, in a typical non-obituary document, this product is zero or very small so that this correlation relation can play an important role in discerning a non-obituary document.

In order to further test the performance of our proposed probabilistic model, we purposely selected five multiple-record Web documents that are similar to multiple-record obituary documents. These documents include a (i) death notice, (ii) bibliography for famous people, (iii) looking for graveyard ads, (iv) list of politicians died youngest, and (v) list of politicians died in car accidents, respectively. Table 4.2 shows the numbers of occurrences of the index terms (i.e., lexical objects) recognized by the

| *Document Region* | *URL* | Probability |
|---|---|---|
| East Coast | http://www.blandnews.com/obit.htm | 1.00 |
| North East Central | http://www.altonweb.com/bb/obituaries/index.html | 1.00 |
| North East Coast | http://rutlandherald.nybor.com/Obits/3327.html | 1.00 |
| North West Central | http://www.bismarktribune.com/tribwebpage/news/ jan2000/1-1obit.html | 1.00 |
| Pacific North West | http://www.thesunlink.com/local/obits/index.html | 1.00 |
| Pacific South West | http://www.honoluluadvertiser.com/obituaries.html | 1.00 |
| South East Central | http://www.andalusiastarnews.com/obituaries/ 2000/01/0112obituaries.html | 1.00 |
| South East Coast | http://www.aikenstandard.com/obits.htnl | 1.00 |
| South West Central | http://www.eurekaherald.com/eureka/obits.html | 1.00 |
| West Coast | http://www.lasvegassun.com/sunbin/stories/archives /1999/oct/28/509480286.html?obituary | 1.00 |

Table 4.4: Positive test Web documents for the obituary ontology

parser along with the density and grouping heuristic values of similar test obituary documents. Most records in this collection include **Deceased Name, Death Date**, and **Birth Date**, and occasionally **Age** and **Interment Address** and there is no occurrences of **Beginning Time**, **Ending Time**, and **Funeral Address**. However, a real obituary document may not include any of these index terms in the worst case. These kinds of obituaries appear in both of our obituary training and test document sets. Based on these observations, we anticipate that our model might have trouble to classify this group of purposely selected documents because they are so close to a real

| *Document Subject* | *URL* | Probability |
|---|---|---|
| Person | http://www.cs.byu.edu/infor/dwembly.html | 1.48e-09 |
| Search Engine | http://www.dogpile.com | 2.67e-10 |
| Purchase Form | http://www.ecampus.com | 1.59e-07 |
| Missing People | http://www.cyberpages.com | 1.43e-04 |
| Jobs | http://www.ohio.com | 2.16e-03 |
| Items for Sale or Rent | http://www.crookstontimes.com | 1.66e-04 |
| Personals | http://www.date-net.com/index.CFM | 6.53e-09 |
| Motorcycles | http://www.netbikes.yks.com/general.html | 0.90 |
| Boats | http://www.sunspot.net/ | 1.57e-06 |
| Antique Cars | http://www.internetclassiccars.com/files<br>/twenties0.html | 3.42e-05 |

Table 4.5: Negative test Web documents for the obituary ontology

| Document | Deceased Name | Age | Death Date | Birth Date | Beginn-ing Time |
|---|---|---|---|---|---|
| Motorcycle | 124 | 1 | 1 | 102 | 2 |
| Obituary | 27 | 37 | 2 | 26 | 12 |
| | Ending Time | Funeral Address | Interment Address | Den-sity | Group-ing |
| Motorcycle | 2 | 101 | 136 | 0.36 | 0.88 |
| Obituary | 7 | 12 | 15 | 0.39 | 0.82 |

Table 4.6: Numbers of occurrences of index terms extracted from the motorcyle document as shown in Figure 4.5 and an obituary document selected from the west coast site

obituary. However, as shown in Table 4.2, only one out of the five documents, i.e., the death notice, causes a false positive. The performance of the our proposed model was re-evaluated by including these similar documents along with the previous twenty test documents: the recall ratio is 100%, the precision ratio drops to $\frac{10}{10+2} = 83.3\%$, and the accuracy ratio reduces to $\frac{25-2}{25} = 92\%$.

| Deceased Name | Age | Death Date | Birth Date | Beginn-ing Time | Ending Time | Funeral Address | Interment Address | Den-sity | Group-ing |
|---|---|---|---|---|---|---|---|---|---|
| 105 | 1 | 76 | 77 | 0 | 0 | 0 | 2 | 0.33 | 0.72 |
| 333 | 0 | 1 | 51 | 0 | 0 | 0 | 3 | 0.26 | 0.35 |
| 436 | 3 | 17 | 36 | 0 | 0 | 0 | 7 | 0.16 | 0.42 |
| 1041 | 0 | 187 | 206 | 0 | 0 | 0 | 1 | 0.18 | 0.52 |
| 433 | 0 | 15 | 49 | 0 | 0 | 0 | 1 | 0.22 | 0.45 |

Table 4.7: Numbers of occurrences of index terms extracted from the five non-obituary but similar test documents

## 4.3 Discussion on the Experimental Results

Our probabilistic model has been tested on both car-ads and obituaries application ontologies. In both applications, false positives occurred during the classification. In the car-ads ontology, one false positive is the boat-snowmobile-for-sale Web document. Figure 4.4 shows a portion of the document and Table 4.3 lists the numbers of occurrences of the index terms extracted from a car-ads document (i.e., the first

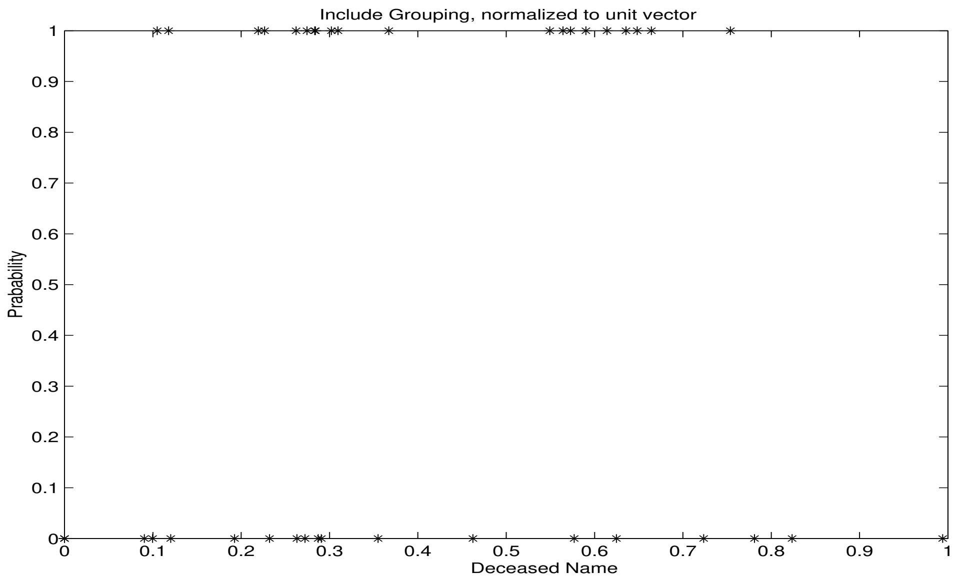



Figure 4.2: Scatter plot of the normalized term frequencies of the index term **Deceased Name** in the obituary ontology

| Document | URL | Probability |
|---|---|---|
| 1996-dead | http://www.dpsinfo.com/dps/1996.shtml | 1.00 |
| biblio | http://www.frii.com/ parrot/biblio.html | 0.00 |
| grave-yard | http://www.findagrave.com/grave/ooooo.html | 0.00 |
| Pldyng | `http://politicalgraveyard.com/index.html#PE` | 0.10 |
| Plcrem | http://politicalgraveyard.com/plcrem.html | 0.00 |

Table 4.8: Test results of our probabilistic model on the five non-obituary but similar test documents

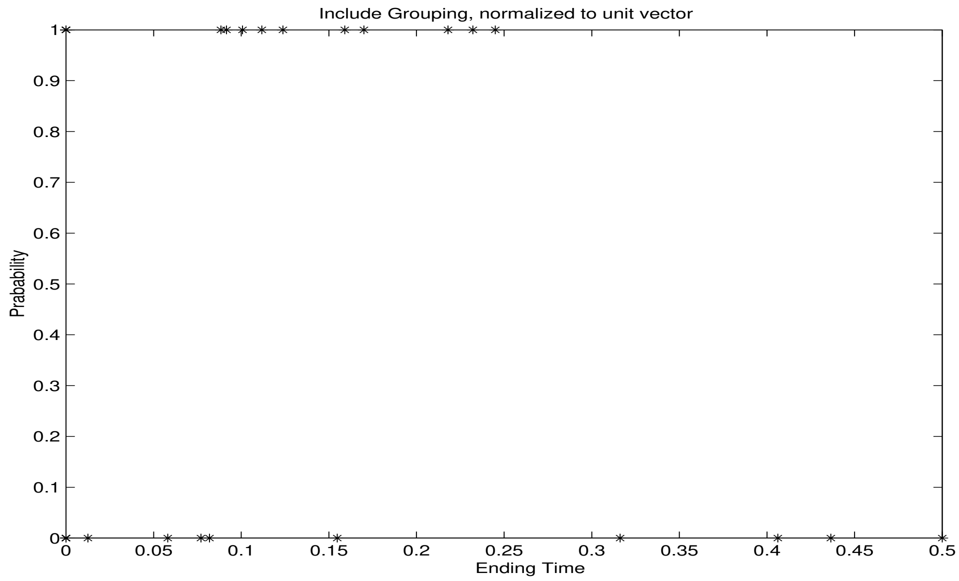



Figure 4.3: Scatter plot of the normalized term frequencies of the index term **Ending Time** in the obituary ontology

document in Table 3.5), a bicycle-for-sale document (i.e., the second document in Table 4.1), and the boat-snowmobile-for-sale document (the third document in Table 4.1). As shown in Figure 4.4, there are several occurrences of **Merc**, **Mercury**, **Jag**, and **Honda** in the boat-snowmobile-for-sale document. These words are interpreted as **Make**. Also, the words **Tracker, Forester**, and **Passport**[2] in the same document are interpreted as **Model**. Furthermore, the density and grouping values in the boat-snowmobile document are (i) much higher than a non car-ads document and (ii) close to the anticipated values appeared in a car-ads document. Using the data in row 23 of Table B.1, we show that **Make, Mileage, density**, and **group-**

[2]The keyword **Forester** does not appear in the page shown in Figure 4.4 due to the limited page size. The whole document can be viewed by following the URL listed in Table 4.1.

## 110-110 BOATS

1974 Glastron boat w/1979 Johnson 35 hp motor, bowmount trolling motor w/2 new batteries, live well & has Shoreland'r trailer, $1,500/OBO. Call 218-736-7939.
1987 16' Lund Predator w/1997 **Mercury** 50 hp., 4 stroke motor, 1987 Roller bunk trailer. Call 218-864-8827.
1988 **Tracker** Magna 17, 100 hp., **Merc** custom trailer, 7.5 **Merc** kicker, custom casting decks, lots of fishing room, $6,500/trade? Call 218-736-5676 or 218-736-0203.
1991 Larson Senza **Merc**. Cruiser, 5.0L motor, EZ loader trailer, must see to appreciate, $9,900/OBO. Call 218-739-6229.
1996 Lund 1775 Pro-V DLX/SE, 2,800 lb. trailer, 150 faststrike Johnson, 9.9 4 stroke Johnson, burnt red/sand, motor colors match, 1999 motor guide 62ES Laser II, 24 volts (new), towbar, rod holders, splash guard, etc., absolute perfect condition, never trailered, 1 lake boat. Call 218-739-6311, after 5pm.
Crestline 14' boat, 50 hp Johnson, trolling motor, depth finder, $1,700/OBO. Call 218-495-2913.

## 110-110 SNOWMOBILES

1980 Arctic Cat **Jag** 3000, very clean, new motor & clutch, cover; 2-place tilt trailer; Empire propane furnace, 125,000 btu. Call 218-495-3667.
1980 Yamaha Enticer DX, rebuilt 340, clutch, new skis, electric start, $650. Call 218-826-6577.
1992 Arctic Cat 440 **Jag** Special, excellent runner, good family sled, $1,400/OBO. Call 218-736-2886.
1994 Polaris 500 EFI, well maintained, very dependable, all new shocks & bearings, 4,300 miles, $1,900. Call 218-583-2366, evenings.
1995 Polaris Indy trail, excellent condition, thumb & hand warmer, $1,800. Call 218-736-7993.
Hitchhiker snowmobile cutter, pull behind seats, (2) $350; also, 1982 **Honda Passport** 70, 1500 miles, excellent, first $500. Call 218-643-4552 or 218-643-9230.

Figure 4.4: Boat-snowmobile-for-sale Web document that causes false positive in the car-ads ontology

**ing** contribute positively to the logodds of the relevance probability. The logodds summation is equal to

$$\begin{aligned}(8.4, 22.8, 61.9, 29.2) \cdot (0.19, 0.30, 0.19, 0.78)^T &\quad + \\ (-1.6, -1.7, -10.1, -20.5) \cdot (1, 1, 1, 1)^T &= 9.07.\end{aligned}$$

The remaining index terms contribute negatively. Their logodds summation is equal to

$$\begin{aligned}(-15.1, 3.7, 15.5, 5.9, -2.5) \cdot (0.71, 0.08, 0.11, 0.05, 0.60)^T &\quad + \\ (8.9, -.9, -3.0, -2.6, 1.1) \cdot (1, 1, 1, 1, 1)^T &= -6.53.\end{aligned}$$

Using the definition of logodds, the relevance probability of the boat-snowmobile document is

$$P = \frac{1}{1 + \exp(-(9.07 + (-6.53)))} = 0.93.$$

This result is computed without considering the correlation relations among the index terms. By adding the correlations, the final probability is equal to 0.84, which is shown in Table 4.1.

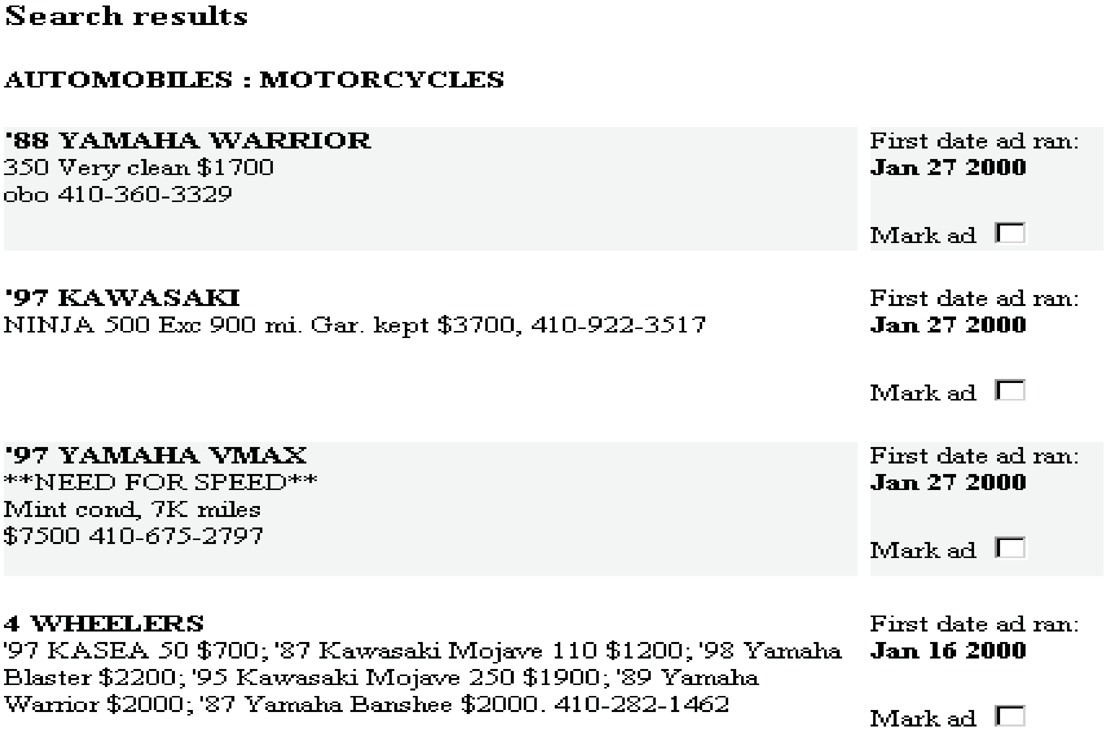
| | Year | Make | Model | Mileage | Price | Feature | Phone | y | z |
|---|---|---|---|---|---|---|---|---|---|
| East Coast | 39 | 37 | 21 | 10 | 34 | 39 | 22 | 0.33 | 0.89 |
| Bicycle | 54 | 2 | 7 | 1 | 4 | 9 | 17 | 0.07 | 0.55 |
| Boat-Snowmobile | 26 | 7 | 3 | 4 | 11 | 2 | 22 | 0.19 | 0.78 |

Table 4.9: Data comparision of a car-ads (East Coast), a non car-ads (Bicycle), and a similar documents (Boat-Snowmobile) with respect to the car-ads application ontology

Using the obituary application ontology on the test set documents, our model yields two false positives. One of the false positive is a **Death notice**, which is very similar to an obituary. The other false positive is caused by the motorcycle-for-sale Web document where a large number of **(Deceased) Name, Birth Date**, and **(Interment) Address** are extracted from the document, and the estimated density and grouping heuristic values of the document are relatively high. As shown in Figure 4.5, besides each record entry, there is an extra box on the right-hand side of each record. Each of these boxes contains the posted date of an advertisement along with a check box “Mark ad”. These extra boxes cause problems. The posted date is treated as **Birth Date** and the word “Mark” in phrase “Mark ad” is misinterpreted as **Deceased Name**. Within each record, the year-make-model combination for a motorcycle is extracted as **Funeral Address**. In addition, there are several records that include multiple motorcycle items. Furthermore, there are multiple occurrences of date that are treated as **Birth Date** and **Death Date**. Since most records have several obituary features mistakenly extracted, our probabilistic model did not classify motorcycle-for-sale document correctly.

The test results for the car-ads and obituary applications are quite surprising. Since the parser for the obituary ontology tends to make more mistakes, we anticipate much better performance of our proposed probabilistic model on the car-ads ontology than the obituary ontology. However, the test results invalidate this assumption. Even though the test results of our probabilistic model could be biased because of the size and the selected documents in the test sets, we believe there are intrinsic reasons for the results. The explanation follows.

The proposed probabilistic retrieval model, which is based on logistic regression, can be viewed as a *weighed majority voting* algorithm. The model determines the relevance probability of a document with respect to an ontology by the contribution of all selected properties (i.e., term frequencies of index terms, the density value, and the grouping value), although they may not contribute evenly. For example, $C_1 = 36.4$



**Search results**

**AUTOMOBILES : MOTORCYCLES**

| | |
|---|---|
| **'88 YAMAHA WARRIOR**<br>350 Very clean $1700<br>obo 410-360-3329 | First date ad ran:<br>**Jan 27 2000**<br>Mark ad |
| **'97 KAWASAKI**<br>NINJA 500 Exc 900 mi. Gar. kept $3700, 410-922-3517 | First date ad ran:<br>**Jan 27 2000**<br>Mark ad |
| **'97 YAMAHA VMAX**<br>**NEED FOR SPEED**<br>Mint cond, 7K miles<br>$7500 410-675-2797 | First date ad ran:<br>**Jan 27 2000**<br>Mark ad |
| **4 WHEELERS**<br>'97 KASEA 50 $700; '87 Kawasaki Mojave 110 $1200; '98 Yamaha Blaster $2200; '95 Kawasaki Mojave 250 $1900; '89 Yamaha Warrior $2000; '87 Yamaha Banshee $2000. 410-282-1462 | First date ad ran:<br>**Jan 16 2000**<br>Mark ad |

Figure 4.5: Motorcycle-for-sale Web document (Follow the URL of the third document in Table 4.1 to view the whole document.) that causes false positive in the obituary ontology

for **Death Date** in the obituary is the largest regression coefficient in the obituary ontology, and hence it carries more weight in the determination of relevance probability of an obituary document than its peer properties as shown in Equation 2.24. If a parser can be directed to detect the most important index terms correctly or to detect most of the index terms correctly, our probabilistic model should have a better chance of correctly categorizing a Web document accordingly. This happens in the obituary ontology. Even though the parser makes mistakes, it can still extracts the important index terms correctly. On the other hand, in the car-ads ontology, our probabilistic model has trouble categorizing similar documents such as snowmobiles and boats-for-sale documents. This is because many index terms in a car-ads document also appear in the closely related documents and some of their domain values are the same. For example, Honda makes cars, motorcycles, and snowmobile. Under this circumstance, our model will have trouble to distinguish the differences. For this reason, the accuracy ratio of our probabilistic model in retrieving relevant documents is not necessarily worse in the obituary ontology than the car-ads ontology.

As mentioned earlier, our probabilistic retrieval model is a weighted majority-voting algorithm. The weight, i.e., the $C_1$ value of each index term, the density value, or grouping value, is determined by the logistic regression analysis on the training document sets for the car-ads and obituary ontologies. Hence, the weights are dependent on the chosen documents in the training set of these ontologies, and their values may be biased if the training document set is too small or not broad enough. For instance, in the obituary ontology, $C_1$ for **Death Date** is much larger than its peers'. This is because in the obituary training document set, only three negative documents include **Death Date** with occurrences of 1, 4, and 1, respectively. On the other hand, nineteen out of 20 positive documents contain large numbers of occurrences of **Death Date**. Hence, **Death Date** becomes a strong dichotomizing factor in determining the relevancy of obituary documents. A subsequent problem of using **Death Date** is that a date with the keyword "death" or "died" nearby it is extracted as the **Death Date**, and the corresponding document is likely to be categorized as an obituaries document. This partially explains why the *Death notice* Web document is a false positive. Contrary to the obituary ontology, the training document set for the car-ads ontology has a better document composition since no regression coefficient is abnormally larger than the others.

As shown in Section 3.5, the inclusion of correlation relations improves the performance of our probabilistic retrieval model. This improvement is expected when we have a better understanding of the roles of our parser and our probabilistic retrieval model. If we have a highly intelligent parser that can extract data correctly, even the most primitive retrieval model could do a good job. However, a real parser, as the one we have been using, makes mistakes, such as splitting a non-multiple-record document into many "records." The desired role of a retrieval model is to make a correct judgment based on the correct information provided by the parser. A good retrieval model should be insensitive to the mistakes made by the parser. However, if a parser makes too many mistakes on each extraction, there is not much a retrieval model can do. Consider Table 3.1 again. We observe that a negative document tends to have the majority of its term frequencies of index terms close to zero. However, in some negative documents, a number of term frequencies of index terms have abnormally large values. This might be either due to mistakes generated by the parser or by the contents of the documents. When we include correlation terms, these abnormal behavior are at least weakened or even can be nullified by other correct information and hence we can improve the immunity of our model to mistakes. However, further study is needed to verify this speculation.

Another contribution of correlation relations to the relevance probability estimation is that they can be used to recognize patterns existing in a document category. Using the obituary application ontology, we can illustrate how a correlation relation is used to enhance the performance of our probabilistic model. Without considering any correlation relation, the relevance probability of a document with respect to an application ontology is estimated based on the accumulated, but independent contributions, from all chosen index terms along with the density and grouping values. A significant number of occurrences of person's name (assumed to be **Deceased Name**) in the document can make a positive contribution to the relevance probability (of an obituary). If the occurrences of **Birth Date** and **Death Date** are detected, the chance that the document is an obituary will be further enhanced. Since all index terms are assumed to make independent contribution, the appearance of **Name-**

**Birth Date** pair, **Name-Death Date** pair, and **Birth Date-Death Date** pair should have similar contribution to the relevance probability of a document with respect to the obituary ontology if they all have comparable term frequencies. However, we must be cautious in using these patterns. If we view these index term pairs as a pattern appeared in an obituary, their existence will definitely have different impact on the relevance probability. For example, a **Name-Birth Date** pair could appear in an obituary or a birth notice. A **Name-Death Date** pair could appear in an obituary or a death notice. However, the appearance of **Birth Date-Death Date**, though not unique to obituaries, will largely enhance the chance to categorize the document to be an obituary document. Hence, by including **Birth Date-Death Date** pattern, i.e., the correlation relation between **Birth Date-Death Date**, along with others, in the relevance probability estimation for the obituary ontology, we improve the performance of our probabilistic retrieval model.

Even though adding correlation relations can improve the model performance, there is a trade-off between the gain of accuracy and the extra computation as well as human interference [BYRN99]. Such human interference includes the extra effort involved in identifying the important correlation relations. Since we use the commercial software to perform logistic regression analysis, the computation of the pairwise correlations needs both human interference (i.e., input data for analysis) and computer time. The complexity involved scales as $O(n^2)$, where $n$ is the number of chosen index terms (i.e., lexical objects) in an ontology. For a complex application ontology, this could add significant computational overhead. An alternative approach of identifying the important correlation relations is by selecting important correlations manually according to the designer's intuition, though not all correlation relations are explicitly identifiable. However, many strong correlation relations could be discovered when we construct an application ontology. Hence, the manual approach is practical when we adopt the proposed probabilistic model based on logistic regression and application ontologies.

# Chapter 5

# Concluding Remarks

We have adopted a binary approach to categorize multiple-record Web documents using a probabilistic retrieval model based on logistic regression analysis and application ontologies. Unlike many other probabilistic retrieval models that use independence assumptions, our probabilistic model tries to weaken the independence assumptions and is capable of analyzing and compensating for existing dependence relations using logistic regression analysis. In this approach, we (1) extract data from training and test documents, (2) apply logistic regression on the training documents, (3) determine which index terms in a training-document set should be included in the relevance probability calculation based on their $p$-values, (4) calculate the relevance probability of each test document with respect to an application ontology based on the regression coefficients on each index term, along with the density and grouping values, in the document computed by the logistic regression analysis, (5) evaluate the performance of our model based on the results from the test-document set, and (6) improve our model performance by using the correlation relations among index terms.

Contrary to the probabilistic retrieval models which use the binary independence assumption [CLvRC98], our probabilistic retrieval model uses the Linked Dependence assumption to weaken the problem of oversimplification caused by the binary independence assumption. In addition, any significant correlation relations among index terms are analyzed and included in our relevance probability estimation.

In order to evaluate the performance of our probabilistic retrieval model in terms of the recall ratio, the precision ratio, and the accuracy ratio, our probabilistic retrieval model has been tested on the car-ads ontology and obituary ontology. For each ontology, the retrieval model is trained on a collection (i.e., a training set) of twenty positive and 20 negative documents. The model performance is then evaluated using different test document sets. The test set for the car-ads ontology contains 28 documents (10 positive and 18 negative documents), whereas the obituary test set contains 25 documents (10 positive and 15 negative documents). To provide a reliable evaluation of our model, we select test documents that are *narrow* and *broad* in content - broad because we want our model to handle different kinds of documents, and narrow because we should be able to catch subtle differences among documents that are similar but belong to different categories. For the car-ads ontology, we achieved a recall ratio of 100%, a precision ratio of 83.3%, and an accuracy of 92.9%. For the obituary ontology, we achieved a recall ratio of 100%, a precision ratio of 83.3%, and an accuracy of 92.0%.

The experimental results show that our probabilistic retrieval model is promising in categorizing multiple-record Web documents with respect to an application ontology. The performance of our model could be better than the above recall, precision, and the accuracy ratios might suggest. This is because a significant portion of the negative test documents are purposely selected such that the structure and contents of these documents are very similar to the corresponding positive documents. Our probabilistic retrieval model is sound, however, we have to rely on commercial statistical software packages to perform logistic regression analysis. Also, scatter plots and human judgments are needed to decide the type of data distribution (e.g., single $S$-curves or double $S$-curves). Hence, the retrieval process is not fully automatic. This tends to be a limitation associated with our model. This, however, is not a major drawback of our proposed model as the limitation can be handled during the process of designing the required ontologies.

# Bibliography


[AA84] A. Albert and J.A. Anderson. On the Existence of Maximum Likelihood Estimates in Logistic Regression Model. *Biometriks*, 71(1):1–10, 1984.

[BJ81] Maurice C. Bryson and Mark E. Johnson. The incidence of monotone likelihood in the cox model. *Technometrics*, 23(4):381–383, November 1981.

[BYRN99] Richardo Baeza-Yates and Berthier Riberro-Neto. *Modern Information Retrieval*. Addison-Wesley, 1999.

[CGD92] Williams S. Cooper, Fredric C. Gey, and Danial P. Dabney. Probabilistic Retrieval Based on Staged Logistic Regression. In *15th Annual Int'l SIGIR*, pages 198–210. SIGIR, 1992.

[CLvR98] Fabio Crestani, Mounia Lalmas, and C.J. van Rijsbergen. *Information Retrieval: Uncertainty and Logics - Advanced Models for the Representation and Retrieval of Information*. Kluwer Academic, 1998.

[CLvRC98] Fabio Crestani, Mounia Lalmas, Cornelis J. van Rijsbergen, and Iain Campbell. Is this document Relevant ... Probably. A survey of Probabilistic Models in Information Retrieval. *ACM Computing Surveys*, 30(4):528–552, December 1998.

[DWHL89] Jr. David W. Hosmer and Stanley Lemesshow. *Applied Logistic Regression*. John Wiley and sons, New York, 1989.

[ECJ+99] D. W. Embley, D. M. Campbell, Y. S. Jiang, W. Liddle, Y.-K. Ng D. W. Lonsdale, and R. D. Smith. Conceptual-model-based Date Extraction from Multiple-Record Web Pages. *Data and Knowledge Engineering*, 31(3):227–251, November 1999.

[EJN99] D.W. Embley, Y.S. Jiang, and Y.-K. Ng. Record-boundary discovery in Web document. In *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data (SIGMOD'99)*, pages 467–478. ACM, June 1999.

[ENX01] D.W. Embley, Y.-K. Ng, and Li Xu. Recognizing Ontology-Applicable Multiple-Record Web Document. To appear in Proceedings of ER'01, November 2001.

[Fuh92] Norbert Fuhr. Probabilistic Models in Information Retrieval. *The Computer Journal*, 35(3):243–255, 1992.

[NWH83] John Neter, William Wasserman, and Michael H. Hutner. *Applied Linear Refression Models*. Richard D. Irwin, INC., 1983.

[Rob77] S.E. Robertson. The probability ranking principle in IR. *Doc.*, 33(4):294–304, 1977.

[vR75] C.J. van Rijsbergen. *Information Retrieval*. Butterworths, 1975.

# Appendix A

# Probability

This appendix section contains some basic results in probability theory, as well as some definitions of terminology. It serves as a reference section.

## A.1 Multiplication Theorem

Let $P(A_1|A_2)$ denote the conditional probability of the occurrence of an event $A_1$, given that event $A_2$ has occurred. If $P(A_2) \neq 0$, then $P(A_1|A_2)$ is evaluated by

$$P(A_1|A_2) = \frac{P(A_1, A_2)}{P(A_2)} \tag{A.1}$$

where $P(A_1, A_2)$ denotes the probability of the occurrence of both events $A_1$ and $A_2$. Equation A.1 is called *multiplication theorem.* Since $P(A_1, A_2) = P(A_2, A_1)$, the direct result of the multiplication theory is

$$\begin{aligned} P(A_1, A_2) = P(A_2, A_1) &= P(A_1|A_2) \cdot P(A_2) \\ &= P(A_2|A_1) \cdot P(A_1). \end{aligned} \tag{A.2}$$

## A.2 Complementary Events

The complementary event of $E$ is denoted as $\bar{E}$. The relationship between $P(E)$ and $P(\bar{E})$ is

$$P(\bar{E}) = 1 - P(E). \tag{A.3}$$

## A.3 Bayes' Rule

Since $P(A_1, A_2) = P(A_2, A_1)$, if both $P(A_1)$ and $P(A_2)$ are not equal to zero, according to Equation A.1,

$$P(A_1|A_2) = \frac{P(A_2, A_1) \cdot P(A_1)}{P(A_2)}, \qquad P(A_2) \neq 0 \quad \text{and} \tag{A.4}$$

$$P(A_2|A_1) = \frac{P(A_1, A_2) \cdot P(A_2)}{P(A_1)}, \qquad P(A_1) \neq 0. \tag{A.5}$$

These relations are called the *Bayes's rule*.

Bayes's rule relates a posterior probability with a prior probability. Using Bayes' rule, $P(A|B)$ can be expressed as

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{A.6}$$

where $P(A)$ is called the *prior probability* because it is updated and adjusted if the sample space changes, whereas $P(A|B)$ is called the *posterior probability* since its new value depends on the updated value of the prior probability. For example, high blood pressure and obesity are closely related. A person with high blood pressure is likely to be overweight, and vice versa. If we let $P(A|B)$ be the probability that a person has high blood pressure given that he is overweight, and let $P(A)$ be the probability that a random chosen American has high blood pressure, and further let $P(B)$ be the probability that a random chosen American is overweight. Then $P(B|A)$ is the probability that a person is overweight given that he has high blood pressure. It is possible that some medical institutes have performed a long-term survey and have reliable statistical estimation for both $P(A|B)$ and $P(A)$. It is safe to assume that these two numbers are relatively stable within a period of time. However, we believe the number of people with obesity is still growing. Hence, the posterior probability $P(B|A)$ must be updated according to the prior probability $P(B)$, which is relatively easy to do. The Bayes's law provides a convenient way to calculate the posterior probability without performing another complicated statistical analysis, and it is the most commonly used method in early information retrieval.

## A.4 Independent Events

Events $A_1$ and $A_2$ are said to be *independent* if

$$P(A_1, A_2) = P(A_1) \cdot P(A_2). \tag{A.7}$$

This concept of independence of two events can be extended to a list of $n$ ($n \geq 1$) events. Hence, given a list of events $A_1, \cdots, A_n$ that are mutually independent

$$P(A_1, \cdots, A_n) = P(A_1) \cdot \cdots \cdot P(A_n). \tag{A.8}$$

# Appendix B

# Data of the Test Set Documents

This appendix section contains the data of the test set documents used in the car-ads and obituaries application ontologies.

## B.1 Data of the Test Set Documents for the Car-ads Ontology

Table B.1 contains the numbers of occurrences of the index terms along with the density and grouping values of the car-ads test set documents, whereas Table B.1 contains the normalized index term frequencies along with the density and grouping values. There are 28 documents in each table. The first 10 rows are for positive test documents, the second 10 are for negative test documents, and the last eight are those that are similar to car-ads documents. The data are extracted from the documents listed in Tables 3.4, 3.4, and 4.1. These documents can be reached by following the URLs listed in the tables.

## B.2 Data of the Test Set Documents for the Obituary Ontology

Table B.2 contains the numbers of occurrences of the index terms along with the density and grouping values of the obituary test set documents, whereas Table B.2 contains the normalized index term frequencies along with the density and grouping values. There are 25 documents in each table. The first 10 rows are for positive test documents, the second 10 are for negative test documents, and the last five are those that are similar to obituaries. The data are extracted from the documents listed in Tables 4.2, 3.4, and 4.2. These documents can be reached by following the URLs listed in the tables.

| Year | Make | Model | Mileage | Price | Feature | Phone | y | z |
|---|---|---|---|---|---|---|---|---|
| 39 | 37 | 21 | 10 | 34 | 39 | 22 | 0.33 | 0.89 |
| 30 | 51 | 63 | 7 | 58 | 26 | 25 | 0.46 | 0.80 |
| 41 | 16 | 12 | 8 | 18 | 24 | 24 | 0.21 | 0.80 |
| 12 | 6 | 6 | 4 | 8 | 18 | 12 | 0.22 | 0.92 |
| 319 | 209 | 265 | 53 | 264 | 214 | 166 | 0.40 | 0.87 |
| 128 | 74 | 44 | 21 | 73 | 116 | 51 | 0.30 | 0.78 |
| 9 | 5 | 9 | 3 | 9 | 11 | 5 | 0.14 | 0.91 |
| 72 | 47 | 48 | 19 | 54 | 216 | 74 | 0.37 | 0.85 |
| 37 | 13 | 15 | 9 | 8 | 25 | 35 | 0.23 | 0.71 |
| 12 | 14 | 12 | 4 | 11 | 22 | 14 | 0.24 | 0.92 |
| 13 | 4 | 1 | 0 | 0 | 0 | 2 | 0.06 | 0.50 |
| 1 | 0 | 0 | 0 | 0 | 3 | 0 | 0.01 | 0.00 |
| 10 | 0 | 2 | 0 | 0 | 0 | 0 | 0.01 | 0.42 |
| 25 | 3 | 3 | 0 | 0 | 29 | 7 | 0.05 | 0.43 |
| 295 | 15 | 63 | 17 | 9 | 10 | 72 | 0.02 | 0.45 |
| 6 | 0 | 0 | 2 | 8 | 0 | 11 | 0.07 | 0.50 |
| 2 | 0 | 10 | 0 | 0 | 20 | 0 | 0.02 | 0.42 |
| 240 | 45 | 8 | 30 | 78 | 25 | 105 | 0.21 | 0.61 |
| 61 | 1 | 7 | 5 | 51 | 12 | 76 | 0.09 | 0.55 |
| 5 | 11 | 6 | 0 | 10 | 15 | 0 | 0.11 | 0.78 |
| 127 | 17 | 14 | 9 | 67 | 18 | 79 | 0.12 | 0.72 |
| 54 | 2 | 7 | 1 | 4 | 9 | 17 | 0.07 | 0.55 |
| 26 | 7 | 3 | 4 | 11 | 2 | 22 | 0.19 | 0.78 |
| 79 | 4 | 2 | 14 | 46 | 12 | 53 | 0.11 | 0.72 |
| 58 | 1 | 13 | 1 | 25 | 10 | 32 | 0.09 | 0.67 |
| 27 | 7 | 6 | 1 | 17 | 9 | 20 | 0.06 | 0.77 |
| 6 | 2 | 2 | 1 | 3 | 5 | 25 | 0.08 | 0.83 |
| 49 | 20 | 6 | 6 | 24 | 26 | 36 | 0.07 | 0.77 |

Table B.1: Numbers of occurrences of the index terms along with the density and grouping values of the car-ads test set documents

| Year | Make | Model | Mileage | Price | Feature | Phone | y | z |
|---|---|---|---|---|---|---|---|---|
| 0.48 | 0.46 | 0.26 | 0.12 | 0.42 | 0.48 | 0.27 | 0.33 | 0.89 |
| 0.28 | 0.46 | 0.57 | 0.06 | 0.53 | 0.24 | 0.23 | 0.46 | 0.80 |
| 0.68 | 0.27 | 0.20 | 0.13 | 0.30 | 0.40 | 0.40 | 0.21 | 0.80 |
| 0.43 | 0.22 | 0.22 | 0.14 | 0.29 | 0.65 | 0.43 | 0.22 | 0.92 |
| 0.53 | 0.35 | 0.44 | 0.09 | 0.44 | 0.36 | 0.28 | 0.40 | 0.87 |
| 0.60 | 0.35 | 0.21 | 0.10 | 0.34 | 0.54 | 0.24 | 0.30 | 0.78 |
| 0.44 | 0.24 | 0.44 | 0.15 | 0.44 | 0.53 | 0.24 | 0.14 | 0.91 |
| 0.28 | 0.18 | 0.19 | 0.07 | 0.21 | 0.85 | 0.29 | 0.37 | 0.85 |
| 0.60 | 0.21 | 0.24 | 0.15 | 0.13 | 0.41 | 0.57 | 0.23 | 0.71 |
| 0.33 | 0.39 | 0.33 | 0.11 | 0.31 | 0.61 | 0.39 | 0.24 | 0.92 |
| 0.94 | 0.29 | 0.07 | 0.00 | 0.00 | 0.00 | 0.15 | 0.06 | 0.50 |
| 0.32 | 0.00 | 0.00 | 0.00 | 0.00 | 0.95 | 0.00 | 0.01 | 0.00 |
| 0.98 | 0.00 | 0.20 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 0.42 |
| 0.64 | 0.08 | 0.08 | 0.00 | 0.00 | 0.74 | 0.18 | 0.05 | 0.43 |
| 0.95 | 0.05 | 0.20 | 0.05 | 0.03 | 0.03 | 0.23 | 0.02 | 0.45 |
| 0.40 | 0.00 | 0.00 | 0.13 | 0.53 | 0.00 | 0.73 | 0.07 | 0.50 |
| 0.09 | 0.00 | 0.45 | 0.00 | 0.00 | 0.89 | 0.00 | 0.02 | 0.42 |
| 0.86 | 0.16 | 0.03 | 0.11 | 0.28 | 0.09 | 0.38 | 0.21 | 0.61 |
| 0.55 | 0.01 | 0.06 | 0.05 | 0.46 | 0.11 | 0.68 | 0.09 | 0.55 |
| 0.22 | 0.49 | 0.27 | 0.00 | 0.44 | 0.67 | 0.00 | 0.11 | 0.78 |
| 0.76 | 0.10 | 0.08 | 0.05 | 0.40 | 0.11 | 0.47 | 0.12 | 0.72 |
| 0.93 | 0.03 | 0.12 | 0.02 | 0.07 | 0.16 | 0.29 | 0.07 | 0.55 |
| 0.71 | 0.19 | 0.08 | 0.11 | 0.30 | 0.05 | 0.60 | 0.19 | 0.78 |
| 0.74 | 0.04 | 0.02 | 0.13 | 0.43 | 0.11 | 0.49 | 0.11 | 0.72 |
| 0.80 | 0.01 | 0.18 | 0.01 | 0.34 | 0.14 | 0.44 | 0.09 | 0.67 |
| 0.68 | 0.18 | 0.15 | 0.03 | 0.43 | 0.23 | 0.50 | 0.06 | 0.77 |
| 0.23 | 0.08 | 0.08 | 0.04 | 0.11 | 0.19 | 0.94 | 0.08 | 0.83 |
| 0.67 | 0.27 | 0.08 | 0.08 | 0.33 | 0.35 | 0.49 | 0.07 | 0.77 |

Table B.2: Normalized term frequencies along with the density and grouping values of the car-ads test set documents

| Deceased Name | Age | Death Date | Birth Date | Beginn-ing Time | Ending Time | Funeral Address | Interment Address | Den-sity | Group-ing |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 3 | 0 | 0 | 0 | 0 | 0.07 | 0.00 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.06 | 0.00 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 0.04 | 0.50 |
| 18 | 4 | 0 | 9 | 0 | 0 | 3 | 25 | 0.23 | 0.59 |
| 122 | 147 | 0 | 7 | 21 | 15 | 0 | 285 | 0.16 | 0.67 |
| 0 | 1 | 0 | 1 | 2 | 0 | 1 | 5 | 0.08 | 0.75 |
| 2 | 4 | 0 | 0 | 0 | 0 | 0 | 25 | 0.08 | 0.39 |
| 124 | 1 | 1 | 102 | 2 | 2 | 101 | 136 | 0.36 | 0.88 |
| 25 | 4 | 0 | 30 | 1 | 0 | 1 | 90 | 0.12 | 0.55 |
| 12 | 1 | 0 | 0 | 0 | 0 | 0 | 15 | 0.38 | 0.54 |
| 6 | 4 | 4 | 5 | 4 | 0 | 1 | 3 | 0.36 | 0.86 |
| 12 | 23 | 25 | 39 | 0 | 0 | 1 | 1 | 0.41 | 0.56 |
| 17 | 22 | 7 | 27 | 17 | 7 | 2 | 12 | 0.24 | 0.71 |
| 0 | 15 | 19 | 3 | 13 | 1 | 0 | 1 | 0.25 | 0.75 |
| 55 | 87 | 66 | 171 | 54 | 14 | 14 | 10 | 0.28 | 0.76 |
| 7 | 24 | 25 | 3 | 40 | 9 | 1 | 4 | 0.39 | 0.76 |
| 6 | 5 | 6 | 5 | 8 | 0 | 0 | 2 | 0.32 | 0.84 |
| 10 | 14 | 13 | 2 | 12 | 3 | 0 | 13 | 0.31 | 0.95 |
| 5 | 4 | 4 | 5 | 0 | 0 | 1 | 0 | 0.50 | 0.81 |
| 27 | 37 | 2 | 26 | 12 | 7 | 12 | 15 | 0.39 | 0.82 |
| 105 | 1 | 76 | 77 | 0 | 0 | 0 | 2 | 0.33 | 0.72 |
| 333 | 0 | 1 | 51 | 0 | 0 | 0 | 3 | 0.26 | 0.35 |
| 436 | 3 | 17 | 36 | 0 | 0 | 0 | 7 | 0.16 | 0.42 |
| 1041 | 0 | 187 | 206 | 0 | 0 | 3 | 1 | 0.18 | 0.52 |
| 433 | 0 | 15 | 49 | 0 | 0 | 0 | 1 | 0.22 | 0.45 |

Table B.3: Numbers of occurrences of the index terms along with the density and grouping values of the obituary test set documents

| Deceased Name | Age | Death Date | Birth Date | Beginning Time | Ending Time | Funeral Address | Interment Address | Density | Grouping |
|---|---|---|---|---|---|---|---|---|---|
| 0.00 | 0.00 | 0.00 | 1.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.07 | 0.00 |
| 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.06 | 0.00 |
| 0.32 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.95 | 0.04 | 0.50 |
| 0.55 | 0.12 | 0.00 | 0.28 | 0.00 | 0.00 | 0.09 | 0.77 | 0.23 | 0.59 |
| 0.35 | 0.43 | 0.00 | 0.02 | 0.06 | 0.04 | 0.00 | 0.83 | 0.16 | 0.67 |
| 0.00 | 0.18 | 0.00 | 0.18 | 0.35 | 0.00 | 0.18 | 0.88 | 0.08 | 0.75 |
| 0.08 | 0.16 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.98 | 0.08 | 0.39 |
| 0.53 | 0.00 | 0.00 | 0.44 | 0.01 | 0.01 | 0.43 | 0.58 | 0.36 | 0.88 |
| 0.25 | 0.04 | 0.00 | 0.31 | 0.01 | 0.00 | 0.01 | 0.92 | 0.12 | 0.55 |
| 0.62 | 0.05 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.78 | 0.38 | 0.54 |
| 0.55 | 0.37 | 0.37 | 0.46 | 0.37 | 0.00 | 0.09 | 0.28 | 0.36 | 0.86 |
| 0.23 | 0.43 | 0.47 | 0.73 | 0.00 | 0.00 | 0.02 | 0.02 | 0.41 | 0.56 |
| 0.38 | 0.49 | 0.16 | 0.60 | 0.38 | 0.16 | 0.04 | 0.27 | 0.24 | 0.71 |
| 0.00 | 0.54 | 0.69 | 0.11 | 0.47 | 0.04 | 0.00 | 0.04 | 0.25 | 0.75 |
| 0.25 | 0.40 | 0.30 | 0.78 | 0.25 | 0.06 | 0.06 | 0.05 | 0.28 | 0.76 |
| 0.13 | 0.44 | 0.46 | 0.06 | 0.74 | 0.17 | 0.02 | 0.07 | 0.39 | 0.76 |
| 0.44 | 0.36 | 0.44 | 0.36 | 0.58 | 0.00 | 0.00 | 0.15 | 0.32 | 0.84 |
| 0.36 | 0.50 | 0.46 | 0.07 | 0.43 | 0.11 | 0.00 | 0.46 | 0.31 | 0.95 |
| 0.55 | 0.44 | 0.44 | 0.55 | 0.00 | 0.00 | 0.11 | 0.00 | 0.50 | 0.81 |
| 0.47 | 0.64 | 0.03 | 0.45 | 0.21 | 0.12 | 0.21 | 0.26 | 0.39 | 0.82 |
| 0.70 | 0.01 | 0.50 | 0.51 | 0.00 | 0.00 | 0.00 | 0.01 | 0.33 | 0.72 |
| 0.99 | 0.00 | 0.00 | 0.15 | 0.00 | 0.00 | 0.00 | 0.01 | 0.26 | 0.35 |
| 1.00 | 0.01 | 0.04 | 0.08 | 0.00 | 0.00 | 0.00 | 0.02 | 0.16 | 0.42 |
| 0.97 | 0.00 | 0.17 | 0.19 | 0.00 | 0.00 | 0.00 | 0.00 | 0.18 | 0.52 |
| 0.99 | 0.00 | 0.03 | 0.11 | 0.00 | 0.00 | 0.00 | 0.00 | 0.22 | 0.45 |

Table B.4: Normalized term frequencies along with the density and grouping values of the obituary test set documents