Chapter 2

LITERATURE SURVEY

Record linkage of files (Fellegi, I P & Sunter, A B 1969) is used to identify duplicates when unique identifiers are unavailable. It relies primarily on matching of names, addresses, and other fields that are typically not unique identifiers of entities. Matching businesses using business names and other information can be particularly difficult (Winkler, W E 1995). Record linkage is also called object identification (Tejada, S et al. 2001; Tejada, S et al. 2002), data cleaning (Do, H H & Rahm, E 2002), approximate matching or approximate joins (Gravano, L et al. 2001; Guha, S et al. 2004), fuzzy matching (Ananthakrishna, R et al. 2002), and entity resolution (Benjelloun, O et al. 2009). Do, H H & Rahm, E 2002 provide an overview of data cleaning and some research problems. (Tejada, S et al. 2001; Tejada, S et al. 2002) showed how to define linkage rules in a database environment. (Hernandez, M A & Stolfo, S J, 1995) gave merge/purge methods for matching in a database situation.

(Sarawagi, S & Bhamidipaty, A 2002) and Winkler (2002) demonstrated the application of machine-learning methods in record linkage situations where training data (possibly small amounts) were available. (Ananthakrishna, R et al. 2002) and (Jin et al. 2002; Jin et al. 2003) provided methods for linkage in very large files in the database environment. (Cohen, W W & Richman, J 2002) showed how to cluster and match entity names using methods that are scalable and adaptive. (Cohen, W W et al. 2003a; Cohen, W W et al. 2003b) provide new methods of adaptive string comparators based on hidden Markov models that improve on the non-adaptive string comparators in a number of situations. (Bilenko, M & Mooney, R J 2003a) provide adaptive, hidden Markov methods that both standardize and parse names and addresses while comparing and matching components of them. (Borkar et al. 2001), (Christen, P et al. 2002), and (Churches, T et al. 2002) use hidden Markov models for adaptive name and address standardization. (Wei, J 2004) provides new Markov edit algorithms that should improve over the algorithms that apply basic hidden Markov models for string comparison.
(Lafferty, J et al. 2001), (McCallum, A & Wellner, B 2003), and (Culotta, A & McCallum, A 2005) used conditional random fields for representing an exceptionally large number of linkage relationships in which the likelihoods are optimized via Markov Chain Monte Carlo (MCMC) and graph partitioning methods. (Ishikawa, H 2003) provides a more general characterization of Markov random fields, graph partitioning, and optimization of likelihoods that may yield faster computational algorithms. (Chaudhuri, S et al. 2005) provide a theoretic characterization of matching in terms of generalized distances and certain aggregates. (Benjelloun, O et al. 2009) provide a characterization of how pairs are brought together during a matching process. In a substantial number of situations, the files are too big to consider every pair in the cross product space of all pairs from two files. (Newcombe, H B & Kennedy, J M 1962; Newcombe, H B 1988) showed how to reduce the number of pairs considered by only considering pairs that agreed on a characteristic such as surname or date-of-birth. Such reduction in the number of pairs is called blocking.

(Hernandez, M A & Stolfo, S J, 1995) also demonstrated the use of multiple passes on a database to bring together pairs. Each pass corresponds to a different sort ordering of the files and pairs are only considered in a sliding window of fixed size. After the pairs are brought together, more advanced, computationally intensive methods are used for comparing them. (McCallum, A et al. 2000) showed that the first step should be a clustering step that is performed with an easily computable, fast method (referred to as canopies) and the second step can use more expensive computational methods. (Chaudhuri, S et al. 2003) showed how to create index structures that allow for certain types of typographical error in matching within a database. In their application, their methods reduced computation by a factor of three in comparison with naïve methods that compare every pair of records in two files.

(Baxter, R et al. 2003) have used a more easily applied method based on q-grams. Still more advanced methods rely on embedding the files and associated string comparators for approximate string comparison in versions of d-dimensional Euclidean space and using sophisticated R-tree bi-comparison searches (Jin, L et al. 2003). With the possibility of significant computation associated with the embedding algorithms, the methods are intended to reduce computation associated with comparing pairs from $O(N^2)$ to $O(N)$ or $O(N\log N)$ where $N$ is the size of the file being searched.
(Yancey, W E & Winkler, W E 2004) have developed the BigMatch technology for matching moderate size lists of 100 million records against large administrative files having upwards of 4 billion records. The methods are faster than the other methods because they rely on models of typographical error corresponding to actual names and addresses that are not always explicitly used in some of the other new methods.

Active Atlas’ (Tejada, S et al. 2001; Tejada, S et al. 2002) consists of two separate components: a candidate generator and a mapping learner. Its goal is to find common entities amongst two record sets from the same domain. The candidate generator proposes a set of potential matches based on the transformations available to the system. The transformation may be one of a number of string comparison types such as equality, substring, prefix, suffix, stemming, or others. These are weighted equally when computing similarity scores for potential matches. Once the candidate generator has completed proposing potential matches, Active Atlas moves on to the second stage. Also it uses the potential matches as the basis for learning mapping rules and transformation weights.

The mapping learner identifies the correct potential matches by adapting the mapping rules and transformations weights to the specific domain. Due to the fact that the initial similarity scores are very inaccurate, the system uses an active learning approach to refine and improve the transformation weights and mapping rules. This approach uses a decision tree committee model for learning with three members in the committee. The mapping learner selects the most informative potential match and asks the user to label this example as either a match or non-match. The user’s response is used to refine and recalculate the transformation weights, learn new mapping rules, and reclassify record pairs. This process continues until: (1) the committee learners converge and agree on one decision tree, or (2) the user has been asked a pre-defined number of questions. Once the mapping rules and transformation weights have been learned, Active Atlas uses them to classify all the potential matches in the system as matched or not matched. The results are then made available to the user.

Fuzzy techniques and methods from information retrieval have been used in some of the systems in the literature to address the data linkage problem. One approach is to represent records as document vectors and compute the cosine distance (Cohen, W 1998) between such vectors.
Another possibility is to use an SQL like language (Galhardas, H, et al. 2000) that allows approximate joins and cluster building of similar records, as well as decision functions that decide the representation of the same entity. Other methods (Maletic, J I & Marcus, A, 2000) include statistical outlier identification; pattern matching, clustering and association rules based approaches. A generic knowledge-based framework based on rules and a Java expert system is presented by (Lee, M L, et al. 2000). The authors also describe the precision-recall dilemma, i.e. choosing a higher recall results in lower precision (more non-duplicates being classified as duplicates), or vice versa.

Some of the researchers have explored the use of machine learning and data mining techniques to improve the linkage process. A hybrid system is described by (Elfeky, M G, et al. 2002) which utilizes both supervised and unsupervised machine learning techniques in the data linkage process. In order to overcome the problem of the lack of availability of training data in real-world data sets, the authors have proposed a hybrid technique where class assignments are made to a sample of the data through clustering, and the resulting data is then used as training set for a supervised classifier (specifically, a decision tree or an instance-based classifier). (Sarawagi, S & Bhamidipaty, A, 2002), apply active learning to the problem of lack of training instances in real-world data. Simply, by repeatedly providing an example which is representative of part of the unclassified data set for clerical review, then using that result to add to the training set of a decision tree classifier, the authors found that review of less than 100 records provided better results than from 7,000 randomly selected reviews.

High-dimensional overlapping clustering (as an alternative to traditional blocking) was used by (McCallum, A, et al. 2000) in order to reduce the number of record pair comparisons to be made, while (Gu, L & Baxter, R, 2004) explored the use of simple clustering together with a user tunable fuzzy region for the class of possible links, allowing user control over the trade-off between accuracy and the amount of clerical review needed.

Another approach was to train distance measures used for approximate string comparisons. (Bilenko, M & Mooney, R J, 2003) have presented a framework for improving duplicate detection using trainable measures of textual similarity.
The authors argue that both at the character and word level, there are differences in importance of certain characters or words. Modifications, like inserts, delete and transpositions, and accurate similarity computations require adapting string similarity metrics for all fields in a database with respect to the particular data domain. Results obtained by the authors on various data sets show that learned edit distance resulted in improved precision and recall results. Very similar approaches are presented by (Nahm, U Y, et al 2002; Yancey, W E, 2004). Combining different learned string comparison methods can result in improved linkage classification as shown by (Cohen, W W, et al. 2003).

Many methods discussed in the literature to match records adopt a framework in which there are two major steps (Koudas, N, et al. 2006; Elmagarmid, A K, et al. 2007) 1. Identifying a similarity function, 2. Matching records. In step 1, the manually labeled & non labeled duplicates records are used with a set of predefined similarity measures, over numeric and string fields identified by domain experts (Hernandez, M A & Stolfo, S J, 1995). They also learned by learning methods like expectation-maximization, (Bilenko, M & Mooney, R J, 2003) decision tree (Cohen, W W & Richman J, 2002), Bayesian Network (Thibaudeau, Y, 1993) or SVM (Winkler, W E, 1988). In step 2, similarity between the candidate pairs is calculated with the help of composite similarity function and highly similar pairs are identified as duplicates.


Other works (Culotta, A & McCallum A, 2005; Dong, X, et al. 2005; Kalashnikov, D V, et al. 2005) address the matching of multiple types of records. The dependencies among multiple record types are not available for many domains. Unsupervised Conflation Method (UCM) is designed for the Web database scenario in which the records to be matched are of a single type with multiple string fields. Also, these records are query dependent. UCM addresses the field weight assignment issues. Similarity measure can be easily incorporated in UCM. It also related to the classification problem using only a single class of training examples.
One-class SVM distinguishes one class of data from another by drawing the class boundary, which needs a huge amount of data to induce the boundary accurately.

Christen, P, 2008 used the true matches or high similarity scores as positive examples, and true mismatches or low similarity scores are used as negative examples. These training examples are used in convergence step to train a classifier, to label the record pairs not present in the training set. Web pages are classified into two stages by learning from positive examples and unlabeled data. During mapping, rule-based classifier is used to collect strong negative examples from the unlabeled data. During convergence, SVM is trained by the positive examples, and the auto-identified negative examples. Then it is used to identify the new negative examples iteratively until it converges. In UCM, weights are adjusted dynamically but in Christen’s method, static weights are used. UCM uses two classifiers, whereas other methods (Yu, H, et al. 2004) use only one classifier during the iterations of the convergence stage. Here, in this stage, it is assumed that most records from the same data source are non-duplicates. Another assumption is the set of positive training examples is correct.

The problem of identifying approximately duplicate records in databases is an essential step for data cleaning and data integration processes. Most existing approaches have relied on generic or manually tuned distance metrics for estimating the similarity of potential duplicates. (Bilenko, M & Mooney, R J 2003), have proposed a framework for improving duplicate detection using trainable measures of textual similarity. It was proposed to employ learnable text distance functions for each database field. We show that such measures are capable of adapting to the specific notion of similarity that is appropriate for the domain of the field. Two learnable text similarity measures suitable for this task are presented: an extended variant of learnable string edit distance, and a novel vector-space based measure that employs a Support Vector Machine (SVM) for training. Experimental results on a range of datasets show that the developed framework can improve duplicate detection accuracy over traditional techniques.

Record deduplication is the task of merging database records that refer to the same underlying entity. In relational databases, accurate deduplication for records of one type is often dependent on the merge decisions made for records of other types. Whereas nearly all-previous approaches have merged records of different types independently, (Culotta, A
& McCallum, A 2005) model these inter-dependencies explicitly to collectively deduplicate records of multiple types. A conditional random field model of deduplication is constructed that captures these relational dependencies, and then employs a novel relational partitioning algorithm to jointly deduplicate records. System is evaluated on two citation matching datasets, for which it is deduplicated. It is shown that by collectively deduplicating paper and venue records, upto 30% error reduction is obtained in venue deduplication. Up to a 20% error reduction is possible in paper deduplication over competing methods.

In the real world, entities have two or more representations in databases. Duplicate records do not share a common key and/or they contain errors that make duplicate matching a difficult task. Errors are introduced as a result of transcription errors, incomplete information, lack of standard formats, or any combination of these factors. (Elmagarmid, A K et al. 2007) have presented a thorough analysis of the literature on duplicate record detection. Also, similarity metrics that are commonly used to detect similar field entries are covered, and an extensive set of duplicate detection algorithms are presented, that can detect approximately duplicate records in a database. Also, multiple techniques for improving the efficiency and scalability of approximate duplicate detection algorithms are covered.

A record-linkage process (Thibaudeau, Y 1993) brings together records from two files into pairs of two records, one from each file, for the purpose of comparison. Each record represents an individual. The status of the pair is a “matched pair” one if the two records in the pair represent the same individual. The status is an “unmatched pair” status if the two records do not represent the same individual. An underlying probabilistic process governs the record-linkage process. A record-linkage rule infers the status of each pair of records based on the value of the comparison. The pair is declared a “link” if the inferred status is that of a matched pair, and it is declared a “non-link” if the inferred status is that of an unmatched pair. The power of discrimination of a record-linkage rule is the capacity of the rule to designate a maximum number of matched pairs as links, while keeping the rate of unmatched pairs designated as links to a minimum. In general, to construct a discriminatory record-linkage rule, some assumptions must be made on the structure of the underlying probabilistic process.
In most of the existing literature, the underlying probabilistic process as an instance of the conditional independence latent class model is assumed. However, this assumption is false in many situations.

In fact, many underlying probabilistic processes do not exhibit the key properties associated with conditional independence latent class models. More general models have been introduced. In particular, latent class models with dependencies are studied and the possibility of improvement in the discrimination power of particular record-linkage rules is shown.

Many commercial organizations routinely gather large numbers of databases for various marketing and business analysis functions. The task is to correlate information from different databases by identifying distinct individuals that appear in a number of different databases typically in an inconsistent and often incorrect fashion. The problem studied is the task of merging data from multiple sources in as efficient manner as possible, while maximizing the accuracy of the result. It is referred to as merge/purge problem. Hernandez, M A & Stolfo, S J 1995) have described the sorted neighborhood method. Some to solve merge/purge uses this and present experimental results that demonstrate this approach may work well in practice but at great cost. An alternative method based upon clustering is also presented with a comparative evaluation to the sorted neighborhood method. It is also shown that a means of improving the accuracy of the results based upon a multi-pass approach. This succeeds by computing the transitive closure over the results of independent runs considering alternative primary key attributes in each pass.

In many data mining projects, information from multiple data sources need to be integrated, combined or linked in order to allow more detailed analysis. The aim of such linkages is to merge all records relating to the same entity, such as patient or a customer. Most of the time, the linkage process is challenged by the lack of a common unique entity identifier, and thus becomes non-trivial. Linking today’s large data collections becomes increasingly difficult using traditional linkage techniques. (Christen, P et al. 2004) have presented an innovative data linkage data systems called Febrl. It includes a new probabilistic approach for improved data cleaning and standardization, innovative indexing methods, a parallelisation approach which is implemented transparently to user,
and a data set generator. It allows the random creation of records containing names and addresses. Implemented, as open source software, Febrl is an ideal experimental platform for new linkage algorithms and techniques.

The task of linking databases is an important step in an increasing number of data mining projects. The reason was linked data can contain information that is not available otherwise, or that would require time-consuming and expensive collection of specific data. The aim of linking is to match and aggregate all records that refer to the same entity. One of the major challenges when linking large databases is the efficient and accurate classification of record pairs into matches and non-matches. While traditionally classification was based on manually set thresholds or on statistical procedures, many of the classification methods then were based on supervised learning techniques. They therefore require training data, which is often not available in real world situations or has to be prepared manually which is an expensive, cumbersome and time-consuming process. A novel, two-step approach to automatic record pair classification has previously been presented by (Christen, P 2008). In its first step, training examples of high quality are automatically selected from the compared record pairs, and used in the second step to train a support vector machine (SVM) classifier. Initial experiments showed the feasibility of this approach, achieving results that outperformed k-means clustering. Two variations of this approach are presented (Christen, P 2008). The first is based on a nearest-neighbor classifier, while the second improves a SVM classifier by iteratively adding more examples into the training sets. Experimental results show that this two-step approach can achieve better classification results than other unsupervised approaches.

U.S. Census Bureau conducted a text census of Tampa, Florida and an independent post enumeration survey (PES) in 1985. The PES was a stratified block sample with heavy emphasis placed on hard-to-count population groups. Matching the individual in the census to the individuals in the PES is an important aspect of census coverage evaluation. Consequently, a very important process for any census adjustment operations that might be planned. For such an adjustment to be feasible, record-linkage software has to be developed that could perform matches with a high degree of accuracy and that is based on an underlying mathematical theory. A principal purpose of the PES is to provide an opportunity to evaluate the newly implemented record-linkage system and the associated methodology.
Jaro, M A, 1989 has discussed the theoretical and practical issues encountered in conducting the matching operations and results are presented. A review of the theoretical background of the record-linkage problem provides a framework for discussions of the decision procedure, file blocking, and the independence assumption.

The estimation of the parameters required by the decision procedure is an important aspect of the methodology, and the techniques presented provide a practical system that is easily implemented. The matching algorithm uses the linear sum assignment model to “pair” the records.

Reference reconciliation (Dong, X et al. 2005) is the problem of identifying when different references (i.e., sets of attribute values) in a dataset correspond to the same real-world entity. Most previous literature assumed references to a single class that had a fair number of attributes (e.g., research publications). Complex information spaces are considered: references taken into account belong to multiple related classes and each reference may have very few attribute values. A prime example of such a space is Personal Information Management, where the goal is to provide a coherent view of all the information on one's desktop. The reconciliation algorithm has three principal features. First, the associations between references to design new methods are exploited for reference comparison. Second, information between reconciliation decisions to accumulate positive and negative evidences was propagated. Third, merging attribute values gradually enriches references. Experiments show the following:

- Precision and recall are improved considerably over standard methods on a diverse set of personal information datasets
- There are advantages to using the developed algorithm even on a standard citation dataset benchmark.

(Kalashnikov, D V et al. 2005) have addressed the problem of reference disambiguation. The situation specifically considered here is that, where entities in the database are referred to using descriptions (e.g., a set of instantiated attributes). The objective of reference disambiguation is to identify the unique entity to which each description corresponds.
The key difference between the proposed approach (called RelDC – Relationships for domain – independent Data Cleaning). The traditional techniques are that RelDC analyzes not only object features but also inter-object relationships to improve the disambiguation quality. Extensive experiments over two real datasets and over synthetic datasets show that analysis of relationships significantly improves quality of the result.

Part of the process of data integration is determining which sets of identifiers refer to the same real-world entities. In integrating databases found on the Web or obtained by using information extraction methods, it is often possible to solve this problem by exploiting similarities in the textual names used for objects in different databases. (Cohen, W W & Richman, J 2002) describe the techniques for clustering and matching identifier names that are both scalable and adaptive, in the sense that they can be trained to obtain better performance in a particular domain. An experimental evaluation on a number of sample datasets shows that the adaptive method sometimes performs much better than either of two non-adaptive baseline systems. Also it is nearly always competitive with the best baseline system.

Many important problems involve clustering large datasets. Although naive implementations of clustering are computationally expensive, there are established efficient techniques for clustering when the dataset has (i) a limited number of clusters, (ii) a low feature dimensionality, or (iii) a small number of data points. However, there has been much less work on methods of efficiently clustering datasets. These are large in all three ways at once – for example, having millions of data points that exist in many thousands of dimensions representing many thousands of clusters. A new technique was presented (McCallum, A et al. 2000) for clustering these large, high dimensional datasets. The key idea involves using a cheap, approximate distance measure to efficiently divide the data into overlapping subsets, called canopies. Then only measuring exact distances between points that occur in a common canopy, the clustering is performed. Using canopies, large clustering problems that were formerly impossible become practical. Under reasonable assumptions about the cheap distance metric, this reduction in computational cost comes without any loss in clustering accuracy. Canopies can be applied to many domains and used with a variety of clustering approaches, including Greedy Agglomerative Clustering, K-means and Expectation-Maximization.
Experimental results on grouping bibliographic citations from the reference sections of research papers are presented.

Here the canopy approach reduces computation time over a traditional clustering approach by more than an order of magnitude and decreases error by 25% in comparison to a previously used algorithm.

Web page classification is one of the essential techniques for Web mining because classifying Web pages of an interesting class is often the first step of mining the Web.

However, constructing a classifier for an interesting class requires laborious preprocessing such as collecting positive and negative training examples. For instance, in order to construct a “homepage” classifier, one needs to collect a sample of homepages (positive examples) and a sample of non-homepages (negative examples). In particular, collecting negative training examples requires arduous work and caution to avoid bias. (Yu, H et al. 2004) present a framework, called Positive Example Based Learning (PEBL), for Web page classification, which eliminates the need for manually collecting negative training examples in preprocessing. The PEBL framework applies an algorithm, called Mapping-Convergence (M-C), to achieve high classification accuracy (with positive and unlabeled data) as high as that of a traditional SVM (with positive and negative data). M-C runs in two stages: the mapping stage and convergence stage. In the mapping stage, the algorithm uses a weak classifier that draws an initial approximation of “strong” negative data. Based on the initial approximation, the convergence stage iteratively runs an internal classifier (e.g., SVM), which maximizes margins to progressively improve the approximation of negative data. Thus, the class boundary eventually converges to the true boundary of the positive class in the feature space.

(Winkler, W E 1988) describes using the EM algorithm for parameter estimation in the Fellegi-Sunter model of record linkage. The method is applicable for more general classes of distributions than those considered by (Fellegi & Sunter 1969). Let $A \times B$ be the product space of two sets $A$ and $B$, which is divided into matches (pairs representing the same entity) and nonmatches (pairs representing different entities). Linkage rules are those that divide $A \times B$ into links (designated matches), possible links (pairs for which a decision was delayed), and nonlinks (designated nonmatches).
Under fixed bounds on the error rates, (Fellegi & Sunter 1969) provided a linkage rule that is optimal in the sense that it minimizes the set of possible links.

The optimality is dependent on knowledge of certain probabilities that are used in a crucial likelihood ratio. In applications, an independence assumption is made that allows estimation of the probabilities. The probabilities are referred to as matching parameters. If the independence assumption is not valid, then linkage rules based on the estimated probabilities may not be optimal.