A Machine Learning approach to Generic Entity Resolution in support of Cyber Situation Awareness

Christopher Moir and Jonathan Dean
Defence Science and Technology Organisation
PO Box 1500, Edinburgh 5111, South Australia

Abstract

This paper introduces the Generic Entity Resolution (GER) framework; a framework that classifies pairs of entities as matching or non-matching based on the entities' features and their semantic relationships with other entities. The GER framework has been developed as part of an AI-based system for the development of Cyber situational awareness and provides a data fusion role by resolving entities discovered across multiple disparate data sources. The approach utilizes supervised machine learning to identify the set of features and semantic relationships that result in the optimum classification accuracy. We evaluated the GER framework using several well-known data sets and compare the framework's accuracy to existing state-of-the-art resolution algorithms. We found that the GER framework's accuracy compares favourably to existing state-of-the-art resolution algorithms for the data sets used in this evaluation.

Keywords: Entity Resolution, Machine Learning, Situation Awareness, Genetic Algorithm

1 Introduction

A much-enhanced cyber situation awareness capability is a priority for Defence in support to cyber warfare (Department of Defence 2009). The widely held definition of situation awareness, as put forward by Endsley (1988), is:

*'the **perception** of the elements in the environment within the volume*

*of time and space, the **comprehension** of their meaning and the*

***projection** of their status in the near future'*

While the definition holds for the cyber domain, generating and maintaining cyber situation awareness is an increasingly challenging task as adversaries become more capable and malware increases in both volume and technical sophistication (Onwubiko and Owens 2012, McAfee Labs 2013, Symantec Corporation 2013). Further, the scope and the complexity of the cyber domain are significantly higher than other domains (McMillan and Tyworth 2012).

Situation awareness in the cyber context has been traditionally generated by a series of techniques such as vulnerability assessment, intrusion detection or digital forensics which are applied at a low-level of data and abstraction (Barford, Dacier et al. 2010). This requires the human operator to develop and maintain the required higher level situation awareness. Commonly this 'picture' is manifested through manual, time-consuming tasks defined by standard operating procedures and kept as a mental model in the analyst's head, aided by tools such as *Security Information and Event Management Systems* (SIEM). This approach is neither scalable nor sustainable due to the complexity of the cyber environment. Cyber situation awareness must be improved for decision makers and considered for automated systems (Blumenthal, Haines et al. 2012).

One of the requirements for the construction of cyber situation awareness (for man or machine) is data fusion from multiple disparate sources (Onwubiko and Owens 2012). This includes entity resolution, which is defined as the process of "*identifying entities (objects, data instances) referring to the same real-world entity*" (Köpcke and Rahm 2010). The process of entity resolution combines multiple observations of an object into a unified representation. Further, the use of a heterogeneous selection of sources, including both typical sources (IDS alerts, network capture and audit logs) with atypical sources (corporate directories, travel documents, business forms) allows for a holistic approach providing a richer representation of entities and a broader context for situational awareness (Grove, Murray et al. 2013).

Performing entity resolution across disparate sources is not trivial. Observations from sensors are not always complete, may not uniquely nor explicitly identify entities present and may come in a variety of formats. Further, rule-based methods require hand-tuning to perform well and are not robust over time (Grove, Murray et al. 2013).

We believe that the development of comprehensive cyber situation awareness will leverage a number of AI and fusion techniques to deal with the volume of data, along with the uncertain, incomplete, erroneous and conflicting nature of information and information sources. In this paper we present one technique known as *generic pair-wise entity resolution* for consideration as part of a broader integrated solution.

Generic pair-wise entity resolution is designed to resolve entities from different sensors with varying attributes and levels of completeness in their representation. Further, our approach differs from other pair-wise techniques as it: firstly, learns which entity features are best for resolving entities; secondly, learns which metrics for a feature result in optimal resolution accuracy; thirdly, utilizes the semantic associations, or relationships, between entities to enhance the accuracy of resolution; and, finally, uses existing machine learning techniques to avoid hand-tuning or utilising domain-specific expert rules, thereby providing a high level of automation to the resolution process.

The underlying assumption of the proposed generic pair-wise entity resolution algorithm is that two instances with similar values for one or more features, or attributes, are more likely to represent the same real-world entity than two records that do not have any values that are the same. If some of these features are the same then the two records may represent matching entities. Conversely, if none of these features are the same then the two records are more likely to represent non-matching entities.

The contributions of this paper are twofold. Firstly, this paper defines the individual processing steps that form our proposed generic entity resolution algorithm. Secondly, this paper evaluates the generic resolution algorithm against existing resolution algorithms using several publically available data sets.

This remainder of this paper is structured as follows: Section 2 discusses existing entity resolution techniques and frameworks; Section 3 describes in detail how the proposed entity resolution framework works; Sections 4 and 5 report on the evaluation of the proposed entity resolution framework using existing data sets; Section 6 provides a discussion on the efficacy and limitations of the proposed entity resolution framework; Sections 7 and 8 contain the conclusions and future work, respectively, for this research.

2 Related Work

The problem of entity resolution is well-studied, and has led to the development of many different algorithms for resolving entities. This section reviews some of these techniques for performing entity resolution, including those that utilize machine learning algorithms to resolve pairs of instances.

The Stanford Entity Resolution Framework (SERF) projects (Benjelloun, Garcia-Molina et al. 2009) have developed a generic resolution framework that focuses primarily on improving the efficiency of entity matching. They were less concerned with how to match instances, instead choosing to focus their attention on developing efficient matching algorithms that minimize the number of comparisons between database records. They defined 3 algorithms for entity resolution: G-Swoosh, R-Swoosh and F-Swoosh. These algorithms are optimized for matching and merging database records. They each make certain assumptions about the record matching and merging operations. The G-Swoosh algorithm is the most general, but least efficient, of all 3 algorithms. The F-Swoosh algorithm can be significantly more efficient than the R-Swoosh algorithm by avoiding repeated feature comparisons (Benjelloun, Garcia-Molina et al. 2009).

Zhao and Ram (2005) propose a multiple classifier system approach that utilizes a variety of supervised machine learning algorithms to resolve records in a database. These algorithms include: neural networks; *k*-nearest neighbours; decision trees; Naïve Bayes; and linear and logistic regression. Individual record fields are compared using exact matching, sub-string matching, Soundex or Levenshtein's string edit distance (Levenshtein 1966). Zhao and Ram's multiple classifier system combines the outputs from multiple individual classifiers to derive an overall resolution. The multiple classifier system uses either bagging (Breiman 1996), boosting (Schapire and Freund 2012) or cross-validated committees (Parmanto, Munro et al. 1995) to combine the outputs from homogeneous base classifiers; for heterogeneous base classifiers it uses cascade generalization (Gama and Brazdil 2000) or stacked generalization (Wolpert 1992).

Bilenko and Mooney's (2003) Multiply Adaptive Record Linkage with Induction (MARLIN) combines multiple string similarity matchers using a Support Vector Machine (SVM), which is a maximal-margin, kernel-based classifier (Cortes and Vapnik 1995). MARLIN utilizes a two level learning approach: at the first level, string similarity measures are trained to estimate the similarity between values in the same database field; at the second level a SVM is trained to identify when two records match using the similarity measures learnt at the first level. MARLIN supports two methods for selecting training data in a semi-automatic manner: static-active and weakly-labelled negative selection. In static-active selection, near duplicate pairs of instances are identified by comparing instances to a string similarity measure and selecting only those pairs that are classified as similar according to this measure. In weakly-labelled negative training selection, MARLIN randomly selects entity pairs that have few shared values as these pairs are least likely to be duplicates. A human operator then verifies each pair is correctly labelled in the set produced by either method.

Christen's (2008) Freely Extensible Biomedical Record Linkage (FEBRL) application allows users to match biomedical records. To match record pairs, the user must first manually select which attributes of the biomedical records to match. FEBRL contains 26 different similarity measures for matching attribute values; most being variations of well-known approximate string comparison algorithms. FEBRL also contains other special functions for comparing numerical values, or fields that contain date, age or time values. The comparison functions all return a similarity value in the range [0, 1], where a score of 0 signifies total dissimilarity between feature values and 1 signifies an exact match. The user-selected attributes form a

vector of similarity scores for each record pair. FEBRL classifies the record pair as either matching or not matching based on the similarity scores.

The Self-Tuning Entity Matching (STEM) (Köpcke and Rahm 2008) framework automatically constructs strategies for matching entities based on their attributes, or features. STEM consists of three principal steps: firstly, the generation of training data; secondly, the computation of attribute similarity for the entity pairs in the training data; and, thirdly, learning the overall entity resolution strategy. STEM combines the output from the individual similarity measures using SVMs, decision trees, logistic regression, or some combination of these 3, to form its entity resolution strategy. An entity pair's overall resolution is determined by the output from STEM's entity resolution strategy.

3 The Generic Entity Resolution Framework

The Generic Entity Resolution (GER) framework is a software architecture for determining whether pairs of entities in data represent the same real-world entity. The GER framework consists of separate software modules that classify an entity pair as matching or not matching by comparing both entities' values for *a single feature, or attribute, only*. Each software module is called a generic feature resolver, or simply a *generic resolver*. The 'first name' generic resolver in Figure 1 classifies two people as matching if it deems their first names the same. If the 'first name' resolver deems the first names different then it will classify the two people as non-matching. The resolvers are termed generic as they are not specific to an individual feature or data type; the generic resolver's operation is the same for every feature and data type.

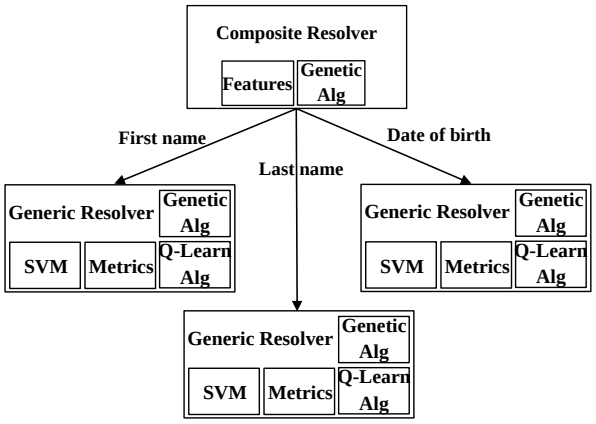

Figure 1: conceptual overview of GER framework illustrating how individual generic resolvers for each comparison feature are combined in a Naïve Bayes network to obtain an overall resolution for pairs of instances

Each generic resolver is comprised of four main components: a set of metrics, which are software functions that map two feature values to a real-valued number in the range [0, 1]; a SVM; a genetic algorithm (Sivanandam and Deepa 2008); and a Reinforcement Learning algorithm known as the Q-learning algorithm (Mitchell 1997). Each function calculates a real-valued number representing a normalized similarity score for the two feature values. The SVM uses these similarity scores to determine the optimum decision boundaries for classifying instances as matching. The genetic algorithm identifies the set of metrics that result in the greatest overall resolution accuracy. Some features can also have more than one value. For example, a person may have multiple variations and spellings of their first name. Each genetic resolver uses the *Q*-learning reinforcement learning algorithm to learn how many values must match before it classifies two instances as matching.

The generic resolvers are combined together to form a naïve version of a Bayesian network (Pearl 1988). The GER framework in Figure 1 resolves two people by comparing their first name, family last name and date of birth. Each comparison feature has its own instance of a generic resolver. Each generic resolver operates independently of the others; the output from one generic resolver does not influence the output from the others.

The root of the Bayesian network, as shown in Figure 1, is a Composite Resolver that classifies two entities as matching or non-matching based on the output from one or more generic resolvers. The Composite Resolver calculates the probabilities that two people match and do not match given the classifications from the first name, last name and date of birth resolver. The Composite Resolver classifies two people as 'matching' if the probability they match exceeds the probability they do not, otherwise it classifies both people as 'not matching'.

The novelty of the GER framework arises from the way it utilizes existing research to resolve entities. The SVM, Q-Learning algorithm, genetic algorithm and Bayesian Network are all used as 'off-the-shelf' components in the GER framework. The GER framework's novelty arises from the way it uses these 'off-the-shelf' components to learn: firstly, how to classify each feature differently from the others; and, secondly, which feature classifications it should use to resolve a pair of instances.

There are two distinct phases in the algorithm for constructing the Bayesian network of generic resolvers. In the first phase, individual generic resolvers are created to classify instances as matching or non-matching based on the values of a single feature. In the second phase, a composite resolver is created to combine the output from one or more generic resolvers to determine an overall classification for pairs of instances. The output from the first phase of the algorithm is a set of generic resolvers that have been optimized for resolving pairs of instances using a single assigned feature only. The output of the second phase of the algorithm is a Bayesian network consisting of one or more generic resolvers that can resolve pairs of instances based on their feature values.

The procedure for creating a set of generic feature resolvers is described in Figure 2. Each feature resolver

utilizes a genetic algorithm to identify the set of metrics that results in the greatest resolution accuracy for the instances of that feature in the validation data. During the optimisation process, the training data is transformed to coordinates using a subset of metrics determined by the genetic algorithm. The SVM is then trained with these coordinates. The SVM's *empirical risk functional*, which is an estimate of the SVM's expected classification error (Vapnik 1995), is calculated using a distinct validation set. The SVM with the lowest empirical risk functional is retained. This process is repeated until the genetic algorithm terminates. Finally, if an instance has more than one value for a feature, such as multiple email addresses or phone number, the Q-learning algorithm is utilized to determine the optimum number of values from the training data that must match to classify pairs of instances as the same.

Constructing a set of generic feature resolvers

Let E be the entity type to resolve, let F be set of features for matching entities, let M_f be the set of metrics for comparing the feature, f:

1. For each $f \in F$:
   a. Create generic resolver, g_f, for f.
   b. Set initial metrics set for g_f to M_f.
   c. Run genetic optimization algorithm for g_f to find optimal metrics set for resolving instances based on f. At each iteration, j, in genetic algorithm:
      i. Convert labelled training data for f to set of labelled feature space coordinates using current evolved metrics set, $M_j \subseteq M_f$.
      ii. Convert labelled independent validation data for f to set of labelled feature space coordinates using metrics set in i).
      iii. Train SVM_j using coordinates from i).
      iv. Classify coordinates from ii) using SVM_j.
      v. Calculate SVM_j empirical risk functional using classifications in iv).
      vi. If empirical risk for SVM_j is lower than a previous SVM_{j-k}, $0 < k < j$, then set optimal SVM and optimal metrics set for g_f to SVM_j and M_j, respectively; otherwise discard SVM_j and M_j.
      vii. Repeat steps i) to vi) until genetic algorithm terminates.
   d. If 1 or more instances of E has more than 1 value for f then use Q-learning algorithm to determine the optimum number of matches for classify instances of E as matching.

Figure 2: the algorithm for constructing a set of generic feature resolvers

In the second phase of the algorithm, the composite resolver identifies the set of features that results in the greatest overall resolution accuracy. The steps for creating the composite resolver are described in Figure 3. The composite resolver learns which features are optimal for resolving pairs of entities from the set of features present in the training data. Instead of arbitrarily choosing the features to resolve pairs of entities, the generic resolver learns the optimal set of features from the training data using a genetic algorithm. The composite resolver combines the output of the generic resolvers for each of the features in the optimal feature set to obtain a single overall classification for a pair of instances.

The GER framework has a library of different metrics, or algorithms, for comparing string values, including: Dice's coefficient; Soundex; Metaphone; Caverphone; regular expression comparison; exact match; Levenshtein; Nysiis; and difflibratio, which is the sequence similarity ratio calculated by the *SequenceMatcher* class in the difflib Python library. These comparison metrics all differ significantly in how they compare strings. For example, the Dice's coefficient algorithm calculates the proportion of bigrams, which are sequences of two adjacent elements in a string, which match in both strings. The Soundex, Metaphone, Nysiis and Caverphone algorithms match words based on their phonetics. These algorithms encode the phonetic sounds in the English language into their own unique intermediate form so that they can match words that sound the same. The regular expression algorithm classifies two strings as matching if both strings fit a specified regular expression pattern. The Levenshtein algorithm calculates the number of insertions, deletions and mutations necessary to convert one word into another. The Levenshtein algorithm deems two words as more alike if fewer transformations are required to convert one string into the other.

Constructing the Bayesian network of generic feature resolvers

Let E be the entity type to resolve, let F be set of features for matching entities, let g_f be the generic resolver for feature, f:

1. Create Composite Resolver, C_E, for E.
2. Set initial feature set for C_E to F.
3. For each $f \in F$:
   a. Add g_f to C_E
4. Run genetic algorithm for C_E to find optimal Bayes Network for resolving instances. At each iteration, j, in genetic algorithm:
   a. Create Bayes Network, BN_j, containing only g_f for features in current evolved feature set, $F_j \subseteq F$.
   b. Convert labelled training data for E to set of labelled feature space coordinates for each $f \in F_j$.
   c. Classify coordinates in 4b) using BN_j.
   d. Calculate BN_j empirical risk using classifications in 4c).
   e. If empirical risk for BN_j is lower than previous BN_{j-k}, $0 < k < j$, then set optimal Bayes network for C_E to BN_j; otherwise discard BN_j.
   f. Repeat steps 4a) to 4e) until genetic algorithm terminates.

Figure 3: the algorithm for creating the composite generic resolver

The GER framework also has metrics defined for comparing dates, time and computer network address. The date and time metrics permit an exact match using the day, month, year, hours, minutes and seconds. The date and time metrics also define a total ordering by virtue of the 'greater than' operator.

Multiple instances of the same *class* of metric were included in the comparison metric library if there were differences in a metric's operation or the metrics were not functionally identical. For example, the Metaphone and Soundex algorithms both belong to the class of phonetic algorithms. Metaphone and Soundex use different sound encodings for phonetic representation. Metaphone represents 'ck' in a word using the letter 'k' while Soundex represents each 'c' and 'k' in a word

using the number 2. Both algorithms were therefore included in the generic resolver's set of candidate metrics due to their markedly different operation. Three different versions of the Soundex algorithm were also included as each version represented their sound encodings slightly differently. Similarly, 'strike a match' and Dice's coefficient were also included due to slight differences in implementation approaches between our algorithms. Since it is not known *a priori* which implementations of a given algorithm will provide better results, it was decided to include them all in the set of comparison metrics and have the genetic algorithm determine which implementation is better for the training data supplied.

The GER framework can also utilize the associations, or semantic relationships, between different entities to help resolve pairs of instances. For example, there is a 'many-to-many' association between movies and actors: many actors appear in a single movie, and a single actor can appear in many different movies. A movie also has its own attributes, including title, viewer advisory rating, synopsis and release date. These attributes may not permit the GER framework to resolve movies accurately, since different movies may have the same viewer advisory rating or release date. The movie title and synopsis may also not prove useful for resolving movies as a movie is sometimes released under another title in other countries. Further, remakes are considered different movies, but often share the same title and synopsis. However, in the first instance a movie will share the same actors, while for the later the movies are unlikely to share the same actors. The inclusion of the associations between a movie and its actors may therefore improve the GER framework's overall resolution accuracy.

We hypothesized it is possible to resolve entities by learning: first, an 'optimal' set of features for resolving the entities; second, an 'optimal' set of similarity metrics for each feature; and, thirdly, the similarities that demarcate two entities that are the same from two entities that are different. If this research hypothesis is true then the GER framework will correctly resolve entity pairs after undergoing an initial training process to learn the set of optimal features, optimal metrics for each feature and the similarities that designate two entities as the same. This prediction presumes that the optimal features, the optimal metrics for each feature, and the similarities that designate entities as the same, are learnable from the training data. To examine this research hypothesis, a series of evaluations was performed, as detailed in Section 4.

4 Methodology

The GER framework was evaluated using data from the Fodor and Zagat restaurant guides, the Canadian Opinion Research Archive (CORA), and the Abt-Buy e-Commerce data set. These data sets were chosen to evaluate the proposed GER framework because: firstly, they capture the imperfections and nuances typical of real-world data; secondly, these two data sets have previously been used to evaluate other entity resolution algorithms and techniques; and, thirdly, Defence owned cyber-related data sources were not releasable, or hampered open publishing of results. The Abt-Buy data set was selected because this data set is challenging to resolve (Köpcke, Thor et al. 2010). The CORA and Abt-Buy data sets both contain instances with missing information; using these data sets therefore permits an evaluation of the GER framework with incomplete data. Evaluating the proposed GER framework using the Restaurant, CORA and Abt-Buy data sets also provides a direct comparison between the GER framework and existing entity resolution algorithms.

The GER framework was also evaluated using data from the IMDB and themoviedb.org motion picture databases. Two different configurations for the GER framework were evaluated. In the first configuration, the GER framework resolved pairs of movie records using only the basic attributes of the movie, such as the title or synopsis. In the second configuration, the GER framework used the basic attributes of the movie and the association between a movie and its actors to determine if two movies matched. The Mann-Whitney U test (Mann and Whitney 1947) was used to determine whether the association between a movie and its actors significantly increased the framework's resolution accuracy.

Separate training, validation and test sets were used to evaluate the GER framework. The SVM kernel function and set of metrics for comparing feature values were learnt from the training data. The validation set was used as a pseudo-test set: during the genetic optimization phase, each SVM was repeatedly evaluated against the validation set to identify which type of SVM kernel and set of metrics produced the greatest resolution accuracy.

A total of 30 evaluations were performed for each of the CORA, Restaurant, Abt-Buy and IMDB-themoviedb.org data sets. New training, validation and test sets were generated for each evaluation. The resolver's F-measure, true positive rate and false positive rate were calculated for each evaluation run; resulting in a sample size of 30 for all three measures. Each evaluation was performed on a Fedora 19 64-bit virtual machine running on an IBM HX5 blade, with 2 Intel Xeon E7-2830 2.13 GHz CPUs, a 100 GB Hard Disk Drive and 110 GB of RAM.

4.1 Data

The Fodor and Zagat restaurant data set consists of 864 records. Restaurants are distinguished by the following four features: name, address, city, and restaurant type. Restaurant telephone numbers were not included in the data set since they are known to artificially 'boost' the resolution accuracy. 112 pairs of records are related to the same restaurants. Figure 4 shows two matching records from the guide. The records for the same restaurant do not match precisely, suggesting that naïve comparison techniques, such as exact string

comparison, are unlikely to accurately resolve pairs of restaurants.

"la cote basque", "60 w. 55th st. between 5th and 6th ave.", "new york", "french"

"la cote basque", "60 w. 55th st.", "new york city", "french (classic)"

Figure 4: matching restaurants from the Restaurant data set

The CORA data set consists of 1295 academic publication citations to 122 computer science research papers. The CORA data set records the following 12 features of a publication: author, volume, title, institution, venue, address, publisher, year, pages, editor, note, and month. Figure 5 shows two matching instances from the CORA data set. The records do not match precisely, suggesting that naïve comparison techniques, such as matching the titles exactly, are unlikely to accurately resolve pairs of publications from the CORA data set.

"kearns, m.","'a bound on the error of cross validation using the approximation and estimation rates, with consequences for training-test split', neural information processing 8,","morgan kaufmann,", "(1996).", "pp. 183-189.", "ed: d.s. touretzky, m,c. mozer and m.e. hasselmo."

"m. kearns.","a bound on the error of cross validation, with consequences for the training-test split.","in advances in neural information processing systems 8.", "the mit press,", "1996.", "to appear."

Figure 5: matching publications from the CORA data set

The Abt-Buy data set consists of over 1000 items for sale at both the Abt.com and Buy.com e-Commerce stores. The Abt.com e-Commerce store records an item's name, description and price; Buy.com records an item's name, description, price and also the manufacturer. Figure 6 shows two matching instances from the Abt-Buy data set. The first instance in Figure 6 has values for the name and manufacturer features, but no values for the price or description features. The second instance in Figure 6 represents the same model of TV. It has values for the name and description features but not for manufacturer or price. These two instances are therefore only comparable by their name.

"Samsung LN32A450 32' 720p LCD HDTV","Samsung"

"Samsung 32' Black Flat Panel Series 4 LCD HDTV - LN32A450","Samsung 32' Black Flat Panel Series 4 LCD HDTV - LN32A450/ 10,000:1 Dynamic Contrast Ratio/ 1366 x 768 True 720p Resolution/ 6ms Response Time/ Cold Cathode Fluorescent Lamp (CCFL)/ Hidden Bottom Speakers/ SRS TruSurround XT/ Built-In ATSC/Clear QAM Tuner/ V-Chip System/ Swivel Stand/ Black Finish"

Figure 6: matching items from the Abt-Buy data set

The IMDB and themoviedb.org databases contain detailed information about movies, the actors, directors and crew. In total, data about 11,992 movies and 56,670 actors were retrieved from the IMDB and themoviedb.org databases. The IMDB and themoviedb.org websites have publically-accessible Application Programming Interfaces (APIs) that allows individuals to retrieve data about movies, the cast of actors, the directors and crew. Both websites format information about a movie using JavaScript Object Notation (JSON).

There are several differences between the data obtained from the IMDB and themoviedb.org databases. The data from themoviedb.org is generally more detailed and contains additional fields than the IMDB data. Figure 7 illustrates this for the movie: 'The Dark Knight'. Each actor in themoviedb.org output has an id, cast id, order and character name compared to just the actor's name from IMDB. The output from themoviedb.org also includes other information not in the IMDB output, such as: the production companies; the movie's budget; and the total movie revenue. The IMDB output includes the languages spoken in the movie which, in the case of 'The Dark Knight', does not exactly match the languages reported by themoviedb.org. The IMDB output also includes fields that are not present in the output from themoviedb.org. For instance, the movie's rating and year of release appear in the IMDB output but not in themovidedb.org output.

{"rating_count": 1002794, "genres": ["Action", "Crime", "Drama", "Thriller"], "rated": "PG-13", "language": ["English", "Mandarin"], "rating": 9.0, "country": ["USA", "UK"], "release_date": 20080718, "title": "The Dark Knight", "year": 2008, "filming_locations": "Times Square, Causeway Bay, Hong Kong", "imdb_id": "tt0468569", "directors": ["Christopher Nolan"], "writers": ["Jonathan Nolan", "Christopher Nolan"], "actors": ["Christian Bale", "Heath Ledger", …], "plot_simple": "When Batman, Gordon and Harvey Dent launch an assault on the mob, they let the clown out of the box, the Joker, bent on turning Gotham on itself and bringing any heroes down to his level.", "runtime": ["152 min"], "type": "M", "also_known_as": ["Batman - El caballero de la noche"]}

{"adult": false, "belongs_to_collection": {"id": 263, "name": "The Dark Knight Collection", "budget": 185000000, "genres": [{"id": 28, "name": "Action"}, {"id": 80, "name": "Crime"}, {"id": 18, "name": "Drama"}, {"id": 53, "name": "Thriller"}], "id": 155, "imdb_id": "tt0468569", "original_title": "The Dark Knight", "overview": "Batman raises the stakes in his war on crime. With the help of Lt. Jim Gordon and District Attorney Harvey Dent, Batman sets out to dismantle the remaining criminal organizations that plague the streets. The partnership proves to be effective, but they soon find themselves prey to a reign of chaos unleashed by a rising criminal mastermind known to the terrified citizens of Gotham as the Joker.", "popularity": 20.37483355, "production_companies": [{"name": "Warner Bros. Pictures", "id": 174}, {"name": "Legendary Pictures", "id": 923}, …], "production_countries": [{"iso_3166_1": "GB", "name": "United Kingdom"}, …], "release_date": "2008-07-18", "revenue": 1001921825, "runtime": 152, "spoken_languages": [{"iso_639_1": "en", "name": "English"}, {"iso_639_1": "zh"}], "status": "Released", "tagline": "Why So Serious?", "title": "The Dark Knight", "vote_average": 7.6, "vote_count": 4452, "casts": {"cast": [{"id": 1810, "name": "Heath Ledger", "character": "Joker", "order": 1, "cast_id": 3, "profile_path": "/azPmwxlJVMVrqImB3rhuWAbrhLy.jpg"}, {"id": 3894, "name": "Christian Bale", "character": "Batman", "order": 0, "cast_id": 35, "profile_path": "/vecCvACI2QhSE5fOoANeWDjxGKM.jpg"}, ...], "crew": [{"id": 525, "name": "Christopher Nolan", "department": "Directing", "job": "Director", "profile_path": "/dZZfxdv6yFqame3K4Y6IT4s17EV.jpg"}, {"id": 527, "name": "Jonathan Nolan", "department": "Writing", "job": "Screenplay"}, ...]}}

Figure 7: Sample JSON output from the IMDB website (top) and themoviedb.org website (bottom)

Not all the available features from the IMDB-themoviedb.org data were used to evaluate the GER framework. The plot description, runtime, release date and actors were the only features used to compare movies. Features such as movie title and IMDB id were not used as these features were deemed too discriminatory; in other words, these features would likely render the resolution too easy. For example, it is possible to perfectly resolve movie instances using just their IMDB id as its value unique to each movie. Excluding movie remakes, most movie titles are unique, so resolving movie instances based on their title is likely to enhance the GER framework's efficacy. Actor instances were compared using only their name as this is the only feature for an actor common to both the IMDB and themoviedb.org data sets.

4.2 Training, validation and test data selection

Distinct training, validation and test sets were constructed from the pre-labelled data in the Restaurant, CORA and Abt-Buy data sets. Records that referred to the same real-world entity (restaurants in the Restaurant data set, publications in the CORA data set and sales items in the Abt-Buy data sets) were constructed by pairing together records with the same label. Records that referred to different real-world entities were constructed by randomly selecting records whose labels did not match. Values for the training, validation and test sets were then randomly selected from the sets of matching and non-matching entity pairs. The training, validation and test sets were not permitted to have duplicate entity pairs. The training, validation and test sets for the Restaurant data contained 112 instance pairs; half labelled as matching and the other half labelled as non-matching. The training and validation sets for the CORA and Abt-Buy data set contained 1000 entity pairs; half were labelled as matching and the other half labelled as non-matching. The test sets for the CORA data set consisted of 200 entity pairs, while the test set for the Abt-Buy data set consisted of 100 entity pairs. The training and validation sets for the IMDB-themoviedb.org evaluation also contained 1000 entity pairs; half matching and the other half non-matching. The IMDB-themoviedb.org test sets also consisted of 200 pairs of instances.

Distinct training, validation and test sets for the IMDB-themoviedb.org evaluation were constructed using a stratified sampling approach, which is the process of dividing members of the population into relatively homogeneous groups and then sampling from these groups individually. Stratified sampling was used to ensure that movie sequels and movies sharing at least one actor were included in the training, validation and test sets. Movies that share at least one actor or are sequels are potentially more difficult for the GER framework to resolve as they are different movies, but have some of the same actors. A sequel often has some, or all, of the same cast, while movies that share at least one actor have at least one association with an actor that is the same. If all the non-matching movie instances did not have any common actors, the GER framework could achieve near-perfect resolution accuracy simply by verifying that the movies had no actors in common. Such a simplistic rule for distinguishing between matching and non-matching movies could potentially result in an overly optimistic assessment of the GER framework's efficacy. A stratified sampling approach helped ensure that some non-matching movies in the training, validation and test sets had the same associations as the matching pairs of movies; thereby providing a less biased evaluation of the genetic resolution framework's efficacy.

5 Results

The GER framework's median F-measure, median true positive rate and median false positive rate for the Restaurant, CORA and Abt-Buy data sets are shown in Table 1. The resolver's median F-measure exceeded 0.96 for all three data sets. A median false positive rate of no more than 0.02 across all three data sets is strong evidence that the GER framework correctly resolved all the non-matching entities in the majority of evaluations. The GER framework's median F-measures for all three data sets suggest that: firstly, nearly all of the resolver's classifications of entities as 'matching' are correct; and, secondly, the resolver correctly resolved nearly all of the matching entities in the majority of evaluations.

Data Set	Median F-measure	Median true positive rate	Median false positive rate
Restaurant	0.963	0.946	0
CORA	0.964	0.940	0.01
Abt-Buy	0.969	0.960	0.02

Table 1: the GER framework's median F-measure, true positive rate and false positive rate for the Restaurant, CORA and Abt-Buy data sets

The distribution of F-measures, true positive rates and false positive rates in Figure 8 suggest that the GER framework is able to accurately resolve entities across multiple test sets. The greater variation in true positive rate for the Restaurant data set highlights that the resolver misclassified more matching entities than non-matching entities in several of the evaluations. A possible explanation for this is some of the matching entity pairs mapped to similar coordinates in feature space as the non-matching entity pairs.

The GER framework favours features that have high discriminatory power. The relative frequency of occurrence for the name and title features for the Restaurant, CORA and Abt-Buy data sets confirm this: the GER framework included these features in every one of the evaluations for these data sets (see Table 2). In 17 out of 30 evaluations the GER framework resolved restaurant pairs using only their names. On the other hand, the GER framework never utilized the address to resolve restaurant instances; nor did it use the editor feature and the manufacturer feature to resolve publications and e-Commerce items, respectively. The editor feature was a poor discriminator of matching and

non-matching publications because different publications may have the same editor. Manufacturer was a poor discriminator of matching and non-matching items because a manufacturer can produce many different items. The restaurant address was a poor discriminator because matching pairs of records in the Restaurant dataset do not specify the restaurant's address identically. For example, the records in Figure 4 a) specify the same address very differently. The GER framework was unable to learn how to accurately distinguish matching and non-matching restaurants from their addresses. As a result, the resolver did not utilize the address feature to resolve pairs of records in the Restaurant dataset.

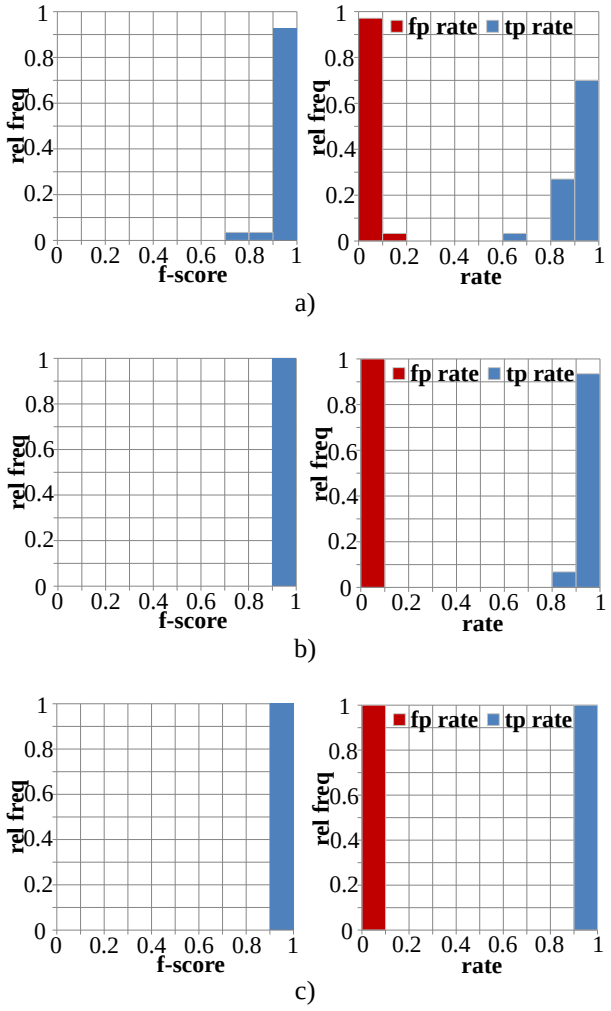

Figure 8: (left) the distribution of F-measures, and, (right) the true and false positive rates for: a) the Restaurant data set; b) the CORA data set; and c) the Abt-Buy data set

Including the association between actors and movies significantly increased the GER framework's overall accuracy for the IMDB-themoviedb.org data sets. Without this movie-actor association the GER framework could only correctly resolve as few as 3 out of 5 and no more than 4 out of 5 pairs of movie instances. The median F-measure increased by approximately 30%, the median true positive rate increased by 42% and the median false positive rate decreased by 3% when the movie-actor associations were included (see Table 3). The Mann-Whitney U test verified that these differences are statistically significant at a 0.05 confidence level. By including the movie-actor associations the GER framework only incorrectly resolved 1 in 20 matching pairs of movies from the IMDB-themoveidb.org databases. The variance in F-measure, true positive rate and false positive rate is also lower with the inclusion of the movie-actor association (see Figure 9); suggesting that the resolution accuracy is consistently greater when the relationship between a movie and its actors is included in the resolution. Given that actors frequently have unique names to disambiguate themselves from others, the inclusion of GER framework that resolves actors based on their name significantly increased the overall resolution accuracy. These results provide empirical evidence that associations between entities (movies and actors in this context) can permit the GER framework to match instances from different data sources more accurately.

Data set	Feature	Relative frequency of feature occurrence in optimal composite resolver
Restaurant	name	1.0
	address	0
	city	0.4
	restaurant type	0.23
CORA	author	0.73
	volume	0.1
	title	1.0
	institution	0.03
	venue	0.6
	address	0.27
	publisher	0.27
	year	0.73
	pages	0.3
	editor	0
	note	0.37
	month	0.1
Abt-Buy	name	1.0
	manufacturer	0
	description	0.6
	price	0.76

Table 2: the proportion of occurrences for each feature in the optimal composite resolver, across all 30 evaluations for each data set

The GER framework's accuracy compares favourably to other entity resolvers' accuracy for the Restaurant and CORA data sets. Bilenko and Mooney (2003) reported maximum F-measures of 0.922 for the Restaurant data set and 0.867 for the CORA data set using their MARLIN system. The GER framework's F-measure is greater than MARLIN's maximum F-measure for both the CORA and Restaurant data sets in over 90% of evaluations. Köpcke and Rahm (2008) attained a maximum F-measure of 0.97 for the Restaurant data set with their STEM algorithm. The GER framework attained a F-measure of at least 0.97 in 13 out of 30 test runs. Chaudhuri, Chen et al. (2007) reported a F-measure of 0.985 for the Restaurant and

CORA data sets using their record matching operator tree algorithm. The authors do not specify whether they repeated their evaluation using different training sets. As a result, it is difficult to assess whether the F-measure of 0.985 is typical for their algorithm. Cohen and Richman (2002) attained maximum F-measure values of 0.964 and 1 for the CORA and Restaurant data sets, respectively. However, their evaluation consisted of only 2 test runs using separate training and test data sets. The GER framework attained a F-measure of *at least* 0.964 in 16 out of 30 evaluations for the CORA data set. In 20 out of 30 evaluations for the Restaurant data set, the GER framework's F-measure is greater than Cohen and Richman's lowest F-measure. Together, these results suggest the GER framework's accuracy for the Restaurant and CORA data sets is similar to existing state-of-the-art resolution algorithms.

The GER framework's accuracy also compares favourably to other entity resolvers' accuracy for the Abt-Buy data set. The GER framework's lowest F-measure score for the Abt-Buy data set is greater than the maximum F-measures reported in Kopcke, Thor et al. (2010) for the same data set. The results reported in Kopcke, Thor et al. also suggest that the GER framework outperforms FEBRL (Christen 2008) and MARLIN (Bilenko and Mooney 2003) for the Abt-Buy data set. Kopcke, Thor et al. (2010) found that FEBRL's and MARLIN's F-measures for the Abt-Buy data set never exceeded 0.8, even for larger training set sizes. The GER framework attained a median F-measure of 0.969 for the same data set. It should be noted that FEBRL and MARLIN only used at most two attributes to resolve instances in the Abt-Buy data set, while the GER framework utilized up to 3 attributes.

Data set	Median F-measure	Median true pos rate	Median false pos rate
Without movie-actor relationship	0.67	0.53	0.03
With movie-actor relationship	0.97	0.95	0.00

Table 3: the GER framework's median F-measure, true positive rate and false positive rate for the IMDB-themoviedb.org data set

A plausible alternative explanation for the GER framework's apparent efficacy is inadvertent inter-dependencies between the training, validation and test sets. Test data that map to the same coordinates in feature space as the training or validation data would most likely introduce optimistic bias in the evaluation. In effect, the GER framework is being evaluated using the same data it was trained on. The GER framework was therefore re-evaluated using training, validation and test data sets that do not map to the same coordinates in feature space.

Re-evaluating the GER framework using CORA test sets that do not map to the same feature space coordinates reduced the framework's median F-measure by 0.03. The F-measures exhibited greater variation for the independent test set, suggesting the framework is unable to resolve pairs of records that represent the same publication as consistently in the independent test set. The Mann-Whitney U test results confirm the framework's efficacy is lower for the independent CORA test sets. The *p*-value for the F-measures using partially dependent and independent test sets was 0.00018, which is statistically significant at $\alpha = 0.05$. Since the median F-measure is lower for the independent test data, it is reasonable to conclude there is an overall reduction in the framework's efficacy when it was evaluated using independent test data. This result is evidence of some optimistic bias in the framework's initial evaluation using the CORA data set. Even accounting for this optimistic bias, the GER framework's precision and recall for the CORA data set still exceeds 90%. It is therefore reasonable to conclude the GER framework's efficacy for resolving the CORA data set is not merely due to test data mapping to the same feature space coordinates as the training or validation data.

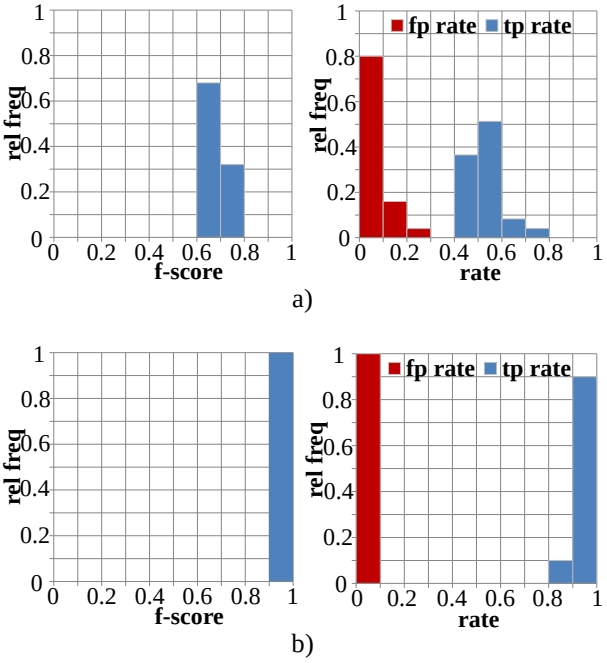

Figure 9: (left) the distribution of F-measures, and, (right) the true and false positive rates for the IMDB-themoviedb.org data set: a) without including actors and b) with actors included in the resolution

Further evidence that the GER framework's efficacy is not solely attributable to optimistic bias was obtained from the results for the Restaurant data set. In one evaluation less than 4% of the test data mapped to the same coordinates as either the training or validation data. The F-measure obtained in this evaluation was 0.943. The F-measure declined by nearly 2% to 0.926 when the duplicate data was removed from the test set. In a second evaluation, less than 8% of the test data mapped to the same coordinates as either the training or validation data. The F-measure fell from 0.982 to 0.935 when the duplicate test data was removed from the test

set; a decrease in F-measure of nearly 5%. Even with the duplicate data removed from both test sets, the GER framework's precision and recall is still high; supporting the claim that the GER framework's efficacy is not solely attributable to any optimistic bias arising from partially dependent test data.

6 Discussion

The experimental results support the research hypothesis that it is possible to resolve instances by learning: the set of features for resolving the entities; the set of similarity metrics for each feature; and the similarities that demarcate matching and non-matching entities. The GER framework's accuracy for the Restaurant, CORA, Apt-Buy and IMDB-themoviedb.org data sets confirms the viability of our proposed machine learning approach to entity resolution. The re-evaluation of the GER framework using independent test data provided further support for the research hypothesis. Even though a slight decline in the resolver's median F-measure was noted with independent test data, the GER framework could still correctly resolve matching and non-matching entity pairs. Together, the experimental results provide strong evidence in support of the research hypothesis.

Examination of the GER framework's output revealed that it failed to correctly resolve CORA citations representing matching publication that humans might also find difficult to resolve. For example, the GER framework failed to correctly resolve the citations in Figure 10 a) and b). The citations in Figure 10 a) represent the same publication even though the publication title and year do not match. The citations in Figure 10 b) also represent the same publication even though the titles differ significantly. It can reasonably be argued that the resolver should classify these citations as different publications since: firstly, different values for the title and year is strong evidence that the publications are different; and, secondly, the same value for the author provides less evidence that the publications are the same since researchers typically publish many papers during their career. The citations in Figure 10 a) provide a clue that they may represent the same publication: the venue is the same for both. This clue alone does not provide overwhelming evidence that the two citations in Figure 10 a) represent the same publication because an author may submit multiple publications to the same academic journal or conference. In sum, some people would classify both pairs of citations in Figure 10 as representing different publications; so it is of little surprise then that the GER framework would also classify both pairs of citations as representing different publications.

The composition of the GER framework is strongly influenced by the feature dependencies entailed in the training set. Figure 11 shows a pair of records from the Restaurant data set that the GER framework incorrectly classified as different restaurants. The only feature that differed in these two records was the restaurant name; all other features were identical in value. It can be argued that both records obviously represent the same restaurant so the GER framework should have resolved them accordingly. Yet following the training phase containing only 112 training instances, the GER framework identified that only the restaurant name and type were needed to resolve pairs of records from the Restaurant data set; with restaurant name given significantly more evidential weight than restaurant type. Stated another way, the GER framework obtained its optimum resolution accuracy for the training data when it matched pairs of records using just the restaurant name and type. Since the two restaurant names in Figure 11 do not match; the GER framework incorrectly classified the two restaurants as non-matching. With more training instances, the GER framework may have learnt that other features, such as the restaurant address and city, are also useful for resolving restaurant instances.

"schapire r.e., freund y., bartlett p., lee w.s.:", "boosting the margin: a new explanation for the effectiveness of voting methods,", "in proceedings of the fourteenth international conference on machine learning,", "morgan kaufmann,", "1997."

"schapire, r. e., freund, y., bartlett, p., & lee, w. s.", "query by committee.", "in proceedings of the 14th international conference on machine learning.", "morgan kaufmann. 205 seung,", "(1992).", "h. s., opper, m., & sompolinsky, h."

a)

"robert e. schapire.", "5(2)", "the strength of weak learnability.", "machine learning,", "1990.", "197-227,"

"r. e. schapire.", "pattern languages are not learnable.", "in proceedings of colt '90,", "morgan kaufmann,", "1990.", "pages 122-129."

b)

Figure 10: 2 pairs of citations in the CORA data set that the GER framework failed to classify as representing the same publication

An advantage of the proposed GER framework compared to other approaches is its reduced reliance on *a priori* knowledge of the data set to determine the optimal model parameters for resolving entities. Köpcke and Rahm (2008) argue there are three key decisions that determine the success of entity resolution: firstly, the selection of features; secondly, the choice of similarity measures; and, thirdly, the selection of similarity threshold values for comparing similarity scores, where the similarity threshold values correspond to the decision boundaries for the SVMs for each feature in the GER framework. If a resolver uses features or metrics that poorly discriminated between matching and non-matching instances its accuracy is likely to decline as a result. If the similarity threshold values are set too high the resolver may classify matching instances as non-matching; if they are set too low the resolver may instead classify non-matching instances as matching. Given the importance of these three key decisions on the resolution accuracy, it is preferable to use machine learning techniques to identify the features, metrics and similarity threshold values that result in optimal resolution accuracy rather than select them using *a priori* knowledge. Utilizing the

machine-learning approach advocated in this paper enables the GER framework to set the features, metrics and similarity threshold values to optimize its resolution accuracy.

"lulu", "816 folsom st.", "san francisco", "mediterranean"
"lulu restaurant-bis-cafe", "816 folsom st.", "san francisco", "mediterranean"

Figure 11: records from the 'Restaurant' data set that the GER framework failed to correctly classify using only the restaurant name and type

The use of Naïve Bayes networks restricts the GER framework's ability to learn the optimal model for resolving instances. Any model that contains conditional dependencies between individual features is not representable by the GER framework. Stated another way, the use of Naïve Bayes networks to combine the output from individual generic resolvers imposes a restriction on the types of models that the GER framework can evaluate. This restriction is a form of representational bias that defines the states in the GER framework's search space (Gordon and Desjardins 1995, Mitchell 1997). It follows that if a particular model is not contained in the GER framework's search space then the framework cannot fit that model to the training data. For example, the GER framework is unable to represent the following probabilistic dependency between publication page numbers and venue: the same publication is unlikely to appear on identical page numbers in 2 different journals or conference proceedings; and publications appearing on different pages in a journal or conference proceedings are also highly unlikely to be the same. It is therefore reasonable to conclude that the GER framework cannot learn the optimal model for training data that has strong conditional dependencies between 2 or more features.

7 Conclusion

The results described in this paper support the claim that the generic pair-wise entity resolution approach can resolve entities from heterogeneous data sources. The GER framework was able to accurately resolve entities from the Restaurant, CORA, Abt-Buy and IMDB-themoviedb.org data sets. These results support the research hypothesis that the generic pair-wise entity resolution approach can enhance cyber situation awareness by learning: firstly, the 'optimal' set of features for resolving instances; secondly, the 'optimal' set of similarity metrics for comparing feature values; and, thirdly, the similarities that constitute matching entity pairs.

8 Future Work

We plan to evaluate the GER framework using other algorithms for combining the output from individual feature resolvers. It was argued in Section 6 that one can view the use of Naïve Bayes to combine individual feature resolvers as a form of representational bias that restricts the hypotheses that the GER framework can form. To avoid this representational bias, and also to assess the impact of this bias on the GER framework's efficacy, future work will investigate alternative algorithms for combining individual feature resolvers. We also intend to integrate more comparison metrics into the GER framework. We conjecture that the accuracy of the GER framework will improve with the inclusion of additional comparison metrics. To test this claim we will re-evaluate the GER framework using the Restaurant, CORA, Abt-Buy and IMDB-themoviedb.org data sets. Finally, the GER framework is capable of supporting a hierarchical naïve Bayes network containing multiple Composite resolvers. This allows relationships between entities to be exploited to support entity resolution. We did not utilise this feature of the GER framework in this study, so future work will seek to evaluate the effectiveness of this.

9 References

Barford, P., M. Dacier, T. G. Dietterich, M. Fredrikson, J. Giffin, S. Jajodia, S. Jha, J. Li, P. Liu and P. Ning (2010). Cyber SA: Situational awareness for cyber defense. Cyber Situational Awareness, Springer**:** 3-13.

Benjelloun, O., H. Garcia-Molina, D. Menestrina, Q. Su, S. Whang and J. Widom (2009). "Swoosh: a generic approach to entity resolution." The VLDB Journal **18**(1): 255-276.

Bilenko, M. and R. J. Mooney (2003). Adaptive duplicate detection using learnable string similarity measures. Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM.

Blumenthal, U., J. Haines, W. Streilein and G. O'Leary (2012). Information Security for Situational Awareness in Computer Network Defense. Situational Awareness in Computer Network Defense: Principles, Methods and Applications, IGI Global**:** 86-103.

Breiman, L. (1996). "Bagging predictors." Machine Learning **24**(2): 123-140.

Chaudhuri, S., B.-C. Chen, V. Ganti and R. Kaushik (2007). Example-driven design of efficient record matching queries. Proceedings of the 33rd international conference on Very large data bases, VLDB Endowment.

Christen, P. (2008). Febrl: a freely available record linkage system with a graphical user interface. Proceedings of the second Australasian workshop on Health data and knowledge management-Volume 80, Australian Computer Society, Inc.

Cohen, W. W. and J. Richman (2002). Learning to match and cluster large high-dimensional data sets for data integration. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. Edmonton, Alberta, Canada, ACM**:** 475-480.

Cortes, C. and V. Vapnik (1995). "Support-vector networks." Machine Learning **20**(3): 273-297.

Department of Defence (2009). Defending Australia in the asia pacific century: Force 2030: Defence white paper Defence. Canberra, Australia, Dept. of Defence.

Endsley, M. R. (1988). Design and evaluation for situation awareness enhancement. Proceedings of the Human Factors and Ergonomics Society Annual Meeting, SAGE Publications.

Gama, J. and P. Brazdil (2000). "Cascade Generalization." Machine Learning **41**(3): 315-343.

Gordon, D. F. and M. Desjardins (1995). "Evaluation and selection of biases in machine learning." Machine Learning **20**(1-2): 5-22.

Grove, D., A. Murray, D. Gerhardy, B. Turnbull, T. Tobin and C. Moir (2013). An Overview of the Parallax BattleMind v1. 5 for Computer Network Defence. Proceedings of the Eleventh Australasian Information Security Conference - Volume 138, Adelaide, Australia, Australian Computer Society, Inc.

Köpcke, H. and E. Rahm (2008). Training selection for tuning entity matching. QDB/MUD.

Köpcke, H. and E. Rahm (2010). "Frameworks for entity matching: A comparison." Data Knowl. Eng. **69**(2): 197-210.

Köpcke, H., A. Thor and E. Rahm (2010). "Evaluation of entity resolution approaches on real-world match problems." Proceedings of the VLDB Endowment **3**(1-2): 484-493.

Köpcke, H., A. Thor and E. Rahm (2010). "Learning-based approaches for matching web data entities." Internet Computing, IEEE **14**(4): 23-31.

Levenshtein, V. (1966). "Binary codes capable of correcting deletions, insertions, and reversals." Soviet Physics Doklady **10**: 707-710.

Mann, H. B. and D. R. Whitney (1947). "On a test of whether one of two random variables is stochastically larger than the other." The annals of mathematical statistics **18**(1): 50-60.

McAfee Labs (2013) "McAfee Threats Report: First Quarter ".

McMillan, E. and M. Tyworth (2012). An Alternative Framework for Research on Situational Awareness in Computer Network Defense. Situational Awareness in Computer Network Defense: Principles, Methods and Applications, IGI Global**:** 71-85.

Mitchell, T. M. (1997). Machine Learning, McGraw-Hill, Inc.

Onwubiko, C. and T. Owens (2012). Situational Awareness in Computer Network Defense: Principles, Methods and Applications, IGI Global.

Parmanto, B., P. W. Munro and H. R. Doyle (1995). Improving committee diagnosis with resampling techniques. Advances in neural information processing systems.

Pearl, J. (1988). Probabilistic reasoning in intelligent systems: networks of plausible inference, Morgan Kaufmann.

Schapire, R. E. and Y. Freund (2012). Boosting: Foundations and Algorithms, The MIT Press.

Sivanandam, S. N. and S. N. Deepa (2008). Introduction to Genetic Algorithms. Berlin, Heidelberg :, Springer Berlin Heidelberg.

Symantec Corporation (2013) "Symantec Internet Security Threat Report 2013." Symantec Internet Security Threat **18**, 1-36.

Vapnik, V. N. (1995). The Nature of Statistical Learning Theory. New York, NY, Springer New York : Imprint: Springer.

Wolpert, D. H. (1992). "Stacked generalization." Neural networks **5**(2): 241-259.

Zhao, H. and S. Ram (2005). "Entity identification for heterogeneous database integration—a multiple classifier system approach and empirical evaluation." Information Systems **30**(2): 119-132.