Finding you on the Internet:

Entity resolution on Twitter accounts and real world people

to obtain the degree of Master of Science

on Tuesday June 18, 2013

Department of Electrical Engineering Mathematics and Computer Science

University of Twente

by

Henry Been

born on March 15, 1985

in Weststellingwerf, The Netherlands

Supervised by:
dr. ir. Maurice van Keulen
dr. Pascal van Eck

Acknowledgments

This master thesis would not have existed for a number of people around me, so there are a lot of people to thank.

I would like to start with thanking my two supervisors Maurice and Pascal. I am very glad I asked them to be my supervisors. They have both contributed tremendously. Maurice as a stakeholder in my research and as an expert in database technology. Pascal for his precise look on matters, for taking that step back and bringing up all those things I have forgotten. Thanks also to Aimee van Wynsberghe. Even though she was formally not one of my supervisors, I have enjoyed working with her and learned a lot about being a responsible engineer. Finally, my contact at the Dutch social investigative authority, Alex van der Werf, needs to be mentioned. Although my work was formally an internal university project, he was very supportive of it and provided me with some great feedback about particular directions of research. It has been a pleasure to work with the four of you.

On a more personal note, I would like to thank all the people who supported me throughout these nine-years-and-ten-months of studying. Two persons I need to mention explicitly: my parents. They have done all they can to support and guide me towards this moment.

Finally, there is one person I need to thank for putting up with me these last few months of my study. Gerja, my girlfriend, has not seen as much of me since January as we both would have liked. There were a few weekends that I kept working in Enschede and we did not see eachother. I thank her for her patience and look forward to moving to Texel as well and the time we are going to spend together.

Henry Been
June 12, 2013

Summary

Over the last years online social network sites [SNS] have become very popular. There are many scenarios in which it might prove valuable to know which accounts on a SNS belong to a person. For example, the dutch social investigative authority is interested in extracting characteristics of a person from Twitter to aid in their risk analysis for fraud detection.

In this thesis a novel approach to finding a person's Twitter account using only known real world information is developed and tested. The developed approach operates in three steps. First a set of heuristic queries using known information is executed to find possibly matching accounts. Secondly, all these accounts are crawled and information about the account, and thus its owner, is extracted. Currently, name, url's, description, language of the tweets and geo tags are extracted. Thirdly, all possible matches are examined and the correct account is determined.

This approach differs from earlier research in that it does not work with extracted and cleaned datasets, but directly with the Internet. The prototype has to cope with all the "noise" on the Internet like slang, typo's, incomplete profiles, etc. Another important part the approach was repetition of the three steps. It was expected that repeating the discovering candidates, enriching them and eliminating false positives will increase the chance that over time the correct account "surfaces."

During development of the prototype ethical concerns surrounding both the experiments and the application in practice were considered and judged morally justifiable.

Validation of the prototype in an experiment showed that the first step is executed very well. In an experiment With 12 subjects with a Twitter account, an *inclusion* of 92% was achieved. This means that for 92% of the subjects the correct Twitter account was found and thus included as a possible match. A number of variations of this experiment were ran, which showed that inclusion of both first and last name is necessary to achieve this high *inclusion*. Leaving out physical addresses, e-mail addresses and telephone numbers does not influence *inclusion*.

Contrary to those of the first step, the results of the third step were less accurate. The currently extracted features cannot be used to predict if a possible match is actually the correct Twitter account or not. However, there is much ongoing research into feature extraction from tweets and Twitter accounts in general. It is therefore expected that enhancing feature extraction using new techniques will make it a matter of time before it is also possible to identify correct matches in the candidate set.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

In the Netherlands there is a vast set of laws that govern social security, with the goal that no one should have to live in poverty. The social security system in the Netherlands is one the most mentioned positive points about their country [38]. Although this system is a highly valued, almost 30% of the Dutch citizens believes that social security systems in the Netherlands are often used in fraudulent ways [38]. Over 90% of the citizens find fraud unacceptable and are in favor of strong enforcement of the law [38]. In the Netherlands, the Inspection for Social Affairs and Employment (Dutch: Inspectie Sociale Zaken en Werkgelegenheid) [ISZW] is responsible for detecting fraud in the social domain.

The last years social investigative authorities gave much attention to strictly enforcing the law. For example, when in 2004 the Dutch municipalities became responsible for the execution of welfare support laws, a country wide initiative [1] was launched at the same time. This initiative intended to facilitate direct and clear communication to welfare support receivers to inform them about their rights, but also their responsibilities, enforcement and penalties. Also, investigative authorities intensified their detection programs.

Detecting of social fraud is mainly the responsibility of the ISZW. The ISZW employs a

[1] Dutch: Hoogwaardig Handhaven

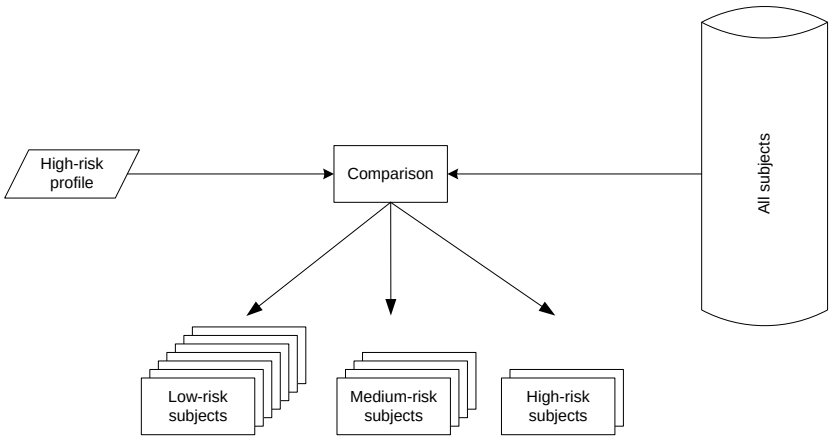


Figure 1.1: Risk analysis: The characteristics of all subjects are compared to a set of high risk characteristics to determine the probability that the subject is fraudulent.

focused approach to this. Every now and then a number of fields where fraud is suspected to be occurring, are chosen to focus on. These fields can be a specific type of fraud or a sector, like welfare fraud, fraud at construction companies or employing illegals. In such a project a team of analysts gathers as much data about all subjects in this field that are eligible for the types of fraud under investigation and perform a risk analysis on all members of this group. The goal of a risk analysis is to identify those subjects that are most likely to be fraudulent.

To perform risk analysis (see also Figure 1.1), first a high risk profile is built. A high risk profile describes a set of characteristics that are commonly shared by a certain type of fraudster. These high-risk profiles are compared with the profile of all subjects to see how many characteristics each subject shares with the high risk profile. The level of similarity between the high-risk profile and the subject his characteristics determines the probability that a subject is a fraudster. Based on this probability subjects are classified as having either a low, middle or high risk of being a fraudster. Based on this classification, physical inspections are performed mainly at subjects who are classified as having a high risk of being a fraudster. Ideally there are only a low number of high-risk subjects so that physical inspections have a high chance of discovering fraud. Determining where to do physical inspections is called signal generation.

In Rotterdam, the Netherlands, it was investigated whether risk analysis was a viable approach to detecting welfare fraud. They concluded that this approach was not successful and did not result in better detection of fraud. They concluded that characteristics of subjects currently available to the authorities, like declared income, country of origin, chamber of commerce records, cannot be used to discriminate between high and low risk subjects. They do speculate that including social characteristics like *attitude to laws* or *values in social network*, might lead to better results [33].

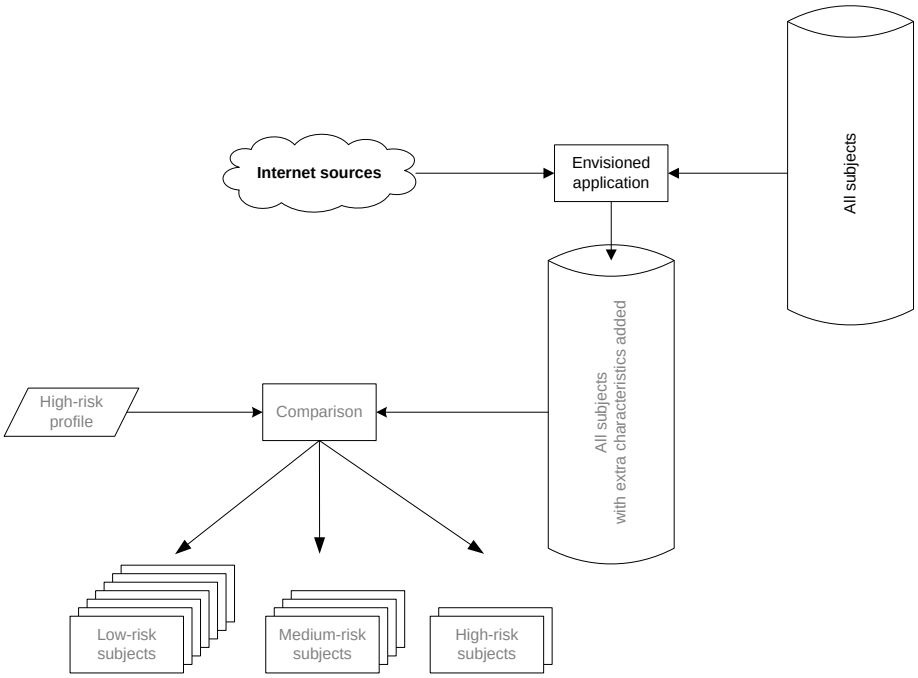


Figure 1.2: Risk analysis using online data: Preprocessing subjects to add more characteristics.

In the Netherlands, investigative authorities are limited in which information they can use to find support receivers that match high risk profiles. Under the Dutch law only limited data can be gathered about all welfare support receivers. Within the ISZW there are people who believe that open data from the Internet can be used to compensate this lack of characteristics of subjects. On the Internet people willingly publish a lot of information about themselves, they make this information intentionally public. In consequence, the ISZW believes that it is not necessarily a privacy infringement to gather and use this data for machine learning purposes.

To facilitate this, first online manifestations of subjects need to be found. In this context, an online manifestation is an online "spot" or collection of web pages where people publish information about themselves. Concrete examples are Facebook, Twitter or LinkedIn profiles. When these online manifestations are found, the information published there needs to be gathered and processed to determine characteristics that can be used in a risk analysis. Currently, he ISZW is investigating if it is possible to develop an application that can perform these tasks and operate as a part their risk analysis. The use of such an application is show in Figure 1.2.

1.2 Problem statement

The solution that the ISZW envisions to solve their lack of discriminating characteristics is an IT system. However, this system still needs to be designed and developed which in turn is an IT problem. A preliminary investigation has shown that there are a number of challenges that need to be overcome, before such a system can be built [4]. However, the most important one is that of online entity resolution. If this problem cannot be solved adequately such a system would not be viable.

Entity resolution is the established field of finding multiple representations of the same object in multiple data sources. For example, when two supermarket chains merge, they might also want to merge the databases of their customer loyalty programs. Since many households have customer loyalty cards of multiple supermarkets, it is likely that some households are in both databases. Entity resolution can then be deployed to identify households that occur in both databases and prevent them from being entered in the new program twice. (For a more elaborate discussion of Entity Resolution see Section 5.2.) Online entity resolution will be defined as the case where there is only one fixed database with entities and the other data source is the Internet.

Online entity resolution has a number of unique aspects. Normally, when performing entity resolution the considered datasets are limited in size. However, when working with the Internet there is a virtually unlimited and fast growing dataset. Also, there is a lot of "noise" on the Internet. Web pages contain typographical errors or mistakes that have to be dealt with. Two measures often used to assess how good something is, are completeness and correctness. These relate directly to the challenges mentioned above. The fact that the Internet is rather large makes it difficult to be complete and even more difficult to validate if the results are complete. The noise on the Internet makes it difficult to deliver correct results.

As stated before, the results of the online entity resolution have to be at least *good enough* to make the application as envisioned by the ISZW viable. In this thesis it is investigated: what is *good enough*, how well can online entity resolution be performed, which factors influence the quality of the results and if this is *good enough*.

1.3 Research Question

The problem statement above leads to the following main research question:

> *Given a limited, and possibly incorrect or incomplete, set of data about a real world person, can their online manifestations be found reliably and automatically?*

This question is divided into six subquestions as follows:

1. Which requirements would investigative authorities, like the ISZW, pose on an application for online entity resolution? And can these requirements be fulfilled?
2. Which data is available on online social networks and other online manifestations?
3. Which data about the real world person is minimally needed to reliably match an online manifestation with a real world person?
4. What is the relation between the time allowed for resolution of entities and the accuracy of the results?
5. Which characteristics of a real world person can be extracted, when these online manifestations are found?

Answering the first subquestion will help to answer the main question with regards for an intended application field. Comparing capabilities to requirements provides perspective and can also be used to judge the viability of applying the researched techniques in a broader context. The second subquestion can be used to identify characteristics that can be gathered online and therefore be used to fuel online entity resolution. The third subquestion is highly related and helps to determine any gap between what can be done and what needs to be done. The questions four and five view the main question from two different points, namely performance when the input data is incorrect or incomplete and the performance over time. Finally, the answer to question six can be used to make some predication about the usefulness of online entity resolution.

1.4 Research Method

The research questions are answered by a combination of interviews, literature research and experimentation with prototypes. The first subquestion will be answered by interviewing employees of the ISZW and comparing his answers to the experimental results. The second question is answered using a combination of literature study and hands-on experience. Literature study will provide general results, while hands-on experience will produce more specific results for the used (the Dutch) population. The third and fourth question can be answered by experimentation with a prototype application that is developed for online entity resolution. Finally, question six can be answered using the gathered data for a qualitative analysis.

In total two experiments with prototypes for online entity resolution are conducted. In the first experiment online entity resolution is performed on real world persons and Twitter accounts. The subjects of this experiment provided a fully informed consent, so the results can be validated. The second experiment is performed on real world persons and Twitter with subjects that have been provided by the ISZW and are the owner or manager of a temporary employment agency. This group is experimented on to investigate to see if the application would perform equally well on persons from another domain and another type of online manifestation. The reason this group of subjects was provided is that the ISZW is currently focusing on this sector in their risk analysis'. Experiment one consists of a number of variations to see if how limiting the information

about the subjects that is used influences the results. Evaluating the results provided after every run will also provide insights into the relation between time elapsed and the quality of the results.

All experiments will be run regularly. In every run the prototype tries to find more possible matches and for each possible match refine the likelihood that it is the correct one. To facilitate this refinement probabilistic database technology is used. These, relatively new, types of databases allow for retaining all possible states of the world and their probability instead of just the one answer.

It is also important to recognize that there are also ethical concerns surrounding this type of research and implementation of the proposed tool. These ethical issues have been considered in collaboration with an ethicist and the results of these considerations are also discussed in this thesis. The main focus of this assessment is finding an appropriate balance between important values like privacy and social security of the Dutch state. Furthermore, a statement is made about the desirability of this type of applications.

1.5 Contributions

1.5.1 Technical contributions

In this report an approach is proposed for finding online manifestations of a real world person: online entity resolution. A prototype, implementing this approach, is designed, implemented and validated. The prototype takes some characteristics of a set of subjects, like first name, last name, e-mail address, etc, and determines what might be their Twitter account. For each possibly matching account a probability of it being correct is also given.

To make this possible a new view on probabilistic databases is developed. This view recognizes three different levels of looking at values: The first level is that of a single value. In certain databases this is the only value, in uncertain databases often the most likely value is given here. The second level is that of possible values and their probabilities. This is the view that probabilistic database up to now take. Finally, a new, the third level is introduced that looks at evidence (support) for each value. Retaining all evidence allows for adjusting the belief in certain values over time due to storing how certain belief levels came to be. It will be shown that a view on each level can be translated to a view of a higher level and that therefore the proposed view is compatible with existing approaches.

1.5.2 Social contributions

The developed prototype serves as a prototype for Dutch investigative authorities. It is expected that investigative authorities take this research and start developing systems for incorporating Internet data into their investigations. This will have benefits for the Dutch society as a whole, it will improve the effectiveness of investigative authorities since their inspectors can be send to inspect, on average, people with higher risks. This is expected to result in the discovery of more fraud and will thus save public money.

Also, the Dutch population does not approve of fraud. Almost all Dutch feel strongly that catching fraudsters is of high importance. Using high quality signal-driven inspection does help to achieve this.

Finally, there is currently no established ethical framework of methodologies or best practices to guide the development of technologies using data from social network sites. The developed prototype was used as a case study to help ethicist develop guidelines for the development of justifiable products that use social network sites.

1.6 Report outline

The next chapter discusses the development of a prototype for online entity resolution. It covers requirements, designing and implementing the prototype. Chapter 3 describes the experiments that were performed and their results. Chapter 4 discusses these results in more detail. Related work is discussed in Chapter 5. In Chapter 6 the ethical aspects of this research are considered. Finally, this thesis ends with a conclusion in Chapter 7.

Chapter 2

Prototype development

2.1 Requirements

As mentioned in Section 1.2 the goal of this master project is to determine if online entity resolution [OER] is viable. To do this a proof-of-concept has to be developed and tested. Before development of such a model can begin, it is important to consider which requirements it should fulfill. And before the requirements for an OER component can be considered, the requirements for the system of which it is a part should be considered. Only when it is known to which requirements the system as a whole has to adhere, requirements on each individual component can be defined. This goes even more for the OER component, since it will be the first component in a series and the other components will have to work with its output. The expectations of other components about their input, or the relation between the quality of their input and output might impose important constrains on the OER component. (See also Figure 1.2 for the different components in an application for risk analysis.) Risk analysis is only one of the methods the ISZW uses for signal generation. For this reason, this section will explore requirements that are posed on signal generation and risk analysis in general and then turn to the OER component are described.

Signal generation is an phenomena for investigative authorities to deploy their resources as effective as possible. For example, at the end of 2012 there were 325.000 people in the Netherlands that received welfare support [9]. It is impossible for the ISZW to do an

in-depth dossier analysis or physical inspection of all these persons to establish if they are really entitled to the support they receive. For this reason, the ISZW recognizes a number of sources of signals that can lead to such an in-depth investigation. Examples include the staff that performs intake interviews with people who applied for welfare support, tips from police of justice departments and, of course, risk analysis.

Table 2.1: Signal generation: Signal quality

Number of signals Number of false positives	High	Low
Low	very valuable	valuable
High	valuable	not valuable

In a report on signal generation [11] the Inspectie Werk en Inkomen, now a part of the ISZW, concludes that there are two ways a source can generate valuable signals. Sources have to have generate either a large quantity of signals or generate few false positives. It follows that a source that generates both a large number of signals and produces few false positive is best, as can also be seen in Table 2.1. Currently, the most valuable source is staff that personally interacts with welfare receivers. Less valuable sources are the department of justice, the chamber of commerce and the police. Currently, risk analysis provides many signals, of which in 15% of the cases fraud is discovered after in-depth analysis.

The most important criteria for information that is used in a risk analysis is correctness of the information used [11]. There is a standard framework, based on the KAD model [13], that the ISZW uses to describe the quality of information. They have derived three main measurements for the quality of data: relevance, reliability and accessibility and can be seen as abstract guidelines.

Relevance concerns the form of the retrieved data. Data should be requested and received in a way that fits with the information need of the annalist that requested the information. The received data should be up to date and not again contain earlier received data. The received data should contain the correct level of detail. The data should be in such a form that it can be compared to other sources. Finally, the form of the data should be such that the way the data is represented can be changed and the data can be altered.

Reliability concerns the extend to which the data can be trusted. First and mostly, data should be correct and complete. Furthermore, data is precise or if numbers are rounded this is noted. The data is neutral and does not reflect a certain bias or opinion. It should be possible to verify the data. Finally, the data is protected against access by anyone except the intended annalist and system failure.

Accessibility concerns the way the data is delivered to the annalist. The annalist should be able to interpret the delivered data, possibly using a supplied explanation. The data should be encoded in such a way that it can be directly used by the ISZW without

involvement of external parties. Finally, the supplier of the information should be available for questions or additional support.

From these general standards, the following requirements can be deduced.

1. The data provided by the OER component should be encoded in XML.
2. The OER component should allow for providing results daily
3. The OER component should achieve a precision of at least 80%.[1]
4. The OER component should achieve as high a recall as possible, without violating the precision requirement
5. The OER component should provide every match with a certainty score: belief, to assess the quality of every individual result.
6. The OER component should not produce biased results, results for males vs. females / foreign vs. native should be the same

XML is a highly standardized way of encoding information. The labeled data fields that XML provide make it easy to interpret and if needed extra information can be added in extra nodes to deliver a self-contained document with both information and description. XML can also be manipulated easily, this makes it possible to select the right level of detail, update data, change the view on the data and make it comparable. This covers both relevance and accessibility from the abstract guidelines.

Reliability is addressed by the requirements 3 to 6. Precision is a statistical measure that describes the percentage of the results that is correct. Recall is a measure that describes the percentage of the good answers that has been delivered. Where precision and recall are calculated for a model as a whole and can describe its overall quality, an individual measure 'belief' or 'trust' should be given for each individual match as well. This measure describes how certain the system is about every individual result and can be used as an *estimation* of the quality of each result. Finally, it is important that the results are not biased. This means that the quality of the results for different groups should be equal or difference only so little that it is statistically insignificant.

[1] What is good enough is hard to establish exactly, but this was mentioned as a guideline by an ISZW data annalist

2.2 Conceptual approach

This section provides a short high-level introduction of the approach taken in this thesis. It also defines some of the terms used through the remainder of the text.

The approach proposed and tested in this thesis consists of three steps. The first step is a search for possibly matching Twitter accounts, using a number of queries. The union of all query results is called the *candidate set*. The most important goal of this first step is achieving a very high likelihood that the correct result is in the *candidate set*. Along with the correct result there may be many incorrect results, called *false positives*. In the second step more details about all items in the *candidate set* are gathered and for each element in the *candidate set* a *similarity score* is calculated. The *similarity score* is a number between 0 and 1 that describes how much the input and each candidate are alike according to some criteria. Per pair, multiple comparison scores can be gathered and eventually combined. The third step is marking false positives, hopefully leaving only one -the correct- result. These three steps together, are called a *run*.

Multiple runs can be executed sequentially over time, with the candidate set being maintained between runs. The expectation is that repeating this process over time will increase the accuracy of the results, for two reasons. The first reason is that query results might sometimes include the correct answer and sometimes not. Repeating the searches over time increases the likelihood that the correct answer is encountered eventually. Secondly, repeating step two allows for enriching all candidates more often and thus improving the accuracy of the comparison functions. For these two reasons, it is expected that the correct Twitter account will "surface" eventually.

This approach can be formalized as follows:

Let P be the set of all persons and T be the set of all Twitter accounts,
then $\forall p \in P$ the correct result $\tilde{t}_p \in T$ is the account of which p is the owner.
(If p has no Twitter account, $\tilde{t}_p$ is not defined)

Let Q be a set of queries q, such that $q : p \to T'$ with $|T'| \ll |T|$ and $0 \ll P(\tilde{t}_p \in T') < 1)$ and R be a set of runs,
then the candidate set for p, for run r is defined as $C_p^r = \bigcup_{q \in Q} (q(p))$

The combined candidate set for p is defined as $C_p = \bigcup_{r \in R} (C_p^r)$
and it follows that the set all false positives is $C_p \backslash \{\tilde{t}_p\}$

Let S be the set of comparison functions s, such that $s : p, t \to [0..1]$,
then the overall similarity of a person p and Twitter account t is given by
$os(p,t) = \sum_{s \in S} s(p,t)$

This leaves us with the task of discovering $w_1 \dots w_n$ such that $os(p, \tilde{t}_p)$ is as close to 1 as possible and $os(p,t)$ (with $t \in T \backslash \{\tilde{t}_p\}$) as close to 0 as possible.

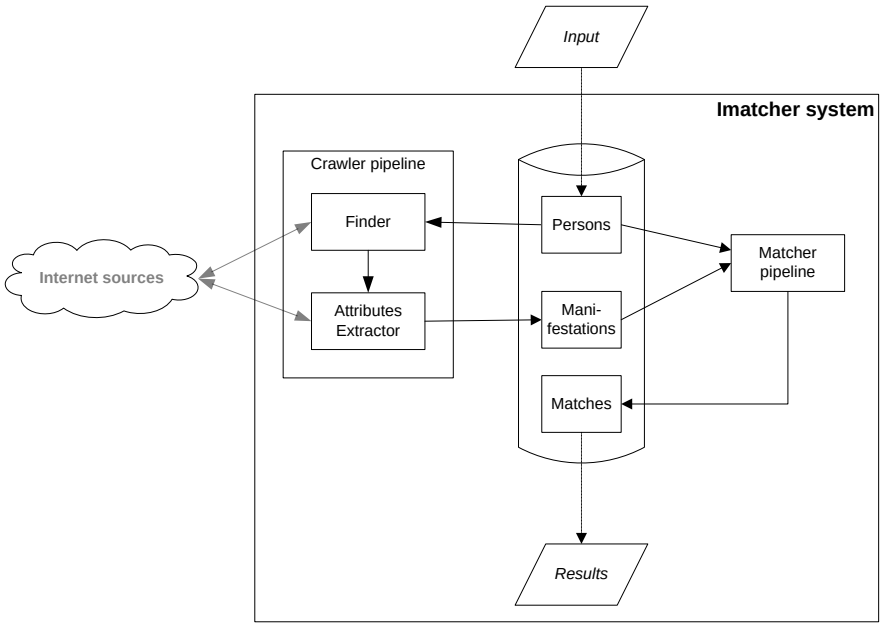


Figure 2.1: IMatcher overall architecture: The pipelines that make up the crawler and matcher are abstracted away.

The following Sections 2.3 and 2.4 describe the design and implementation of the IMatcher. This Java program is the implementation of this algorithm that collects the candidate set for a person and executes all comparision functions. The results of the IMatcher are then used as the input for a classifier in section 3.1.3 that is concerned with discovering the overall similarity function *os* so that it can predict if a given $t \in T$ is $\tilde{t}_p$ for a given person p

2.3 Architectural design

As mentioned in the introduction, online entity resolution has some unique characteristics, compared to regular entity resolution. In regular entity resolution (See Section 5.2 for a more detailed discussion of entity resolution.) there are two (or more) well defined datasets on which the resolution has to be performed. In the given case this characteristic doesn't hold. Only one dataset is given as the input and the other datasets are somewhere on the Internet, they have to be discovered by the application on its own. This results in two tasks for the application to be developed. First, for each data source, candidate matches have to be found and their attributes have to be extracted. This is referred to as crawling. Secondly, all candidate matches have to be compared to the original input to see which matches are the correct ones. This is entity resolution.

After at least one crawl session, entity resolution can be performed on the input data

and the discovered entities. This introduces a second difference with regards to regular entity resolution. Ordinarily, resolution is only performed once, since the datasets on which resolution is performed do not change. In this case, the datasets do change: new candidates and/or new attributes can be discovered in any crawl session. This introduces the need to perform resolution more than once to refine and correct the results.

Both main tasks, crawling and resolution, can be divided into many sequential steps that have to be performed in order to perform the task. An architecture that fits sequential processing by many independent components very well is a pipeline (also known as pipe-and-filters) architecture [16]. A pipeline is formed by a number of sinks that are linked in a producer/consumer fashion. In such an architecture, elementary building blocks called *sinks* are responsible for performing a single task on an elementary piece of data. Each sink provides output that serves as input for the next sink. This approach does not only fit sequential processing very well, it also allows for quickly changing the configuration of a pipeline to include or exclude specific tasks.

In the rest of this thesis, the application under development will be referred to as the IMatcher. In Figure 2.1 the architectural overview of the IMatcher is given. This figure shows a database at the core of the IMatcher. Initially this database is filled with only input data: Persons. The input consist of as much details about the persons that are searched online. The required input is described in Section 2.3.3. Besides the database, there are two main components: the crawler and the matcher. They are responsible for performing the two tasks mentioned before. The matcher operates only on the local database, whereas the crawler also uses the Internet extensively.

The following two sections discuss the overall pipeline architecture of these two parts of the IMachter. Section 2.4.4 provides an overview of all sinks and the implementation details for all sink are provided in Appendix A.

2.3.1 Crawling

In Figure 2.2 the pipelines that make up the crawler are shown. Please note, that although only Twitter related sinks are shown, more sinks and pipelines can exist to explore other data sources. As can be seen on the vertical axis in Figure 2.2, sinks take the same type as both input and output. However, some sinks are are capable of producing extra output, that should serve as the input for another pipeline. This output is shown horizontally and then again processed vertically in another pipeline.

In the current configuration, the first pipeline is for persons. In the IMatcher all input persons are enumerated and fed into the pipeline. As can concluded from the names of the different sinks, this pipeline is responsible for finding possibly matching online manifestations for the given person using different search criteria. An extra sink is added to enumerate already know possible matching manifestations to make sure they are produced as side-output as well. This is to facilitate the second pipeline. This second pipeline is responsible for extracting attributes of the provided manifestation. Even when

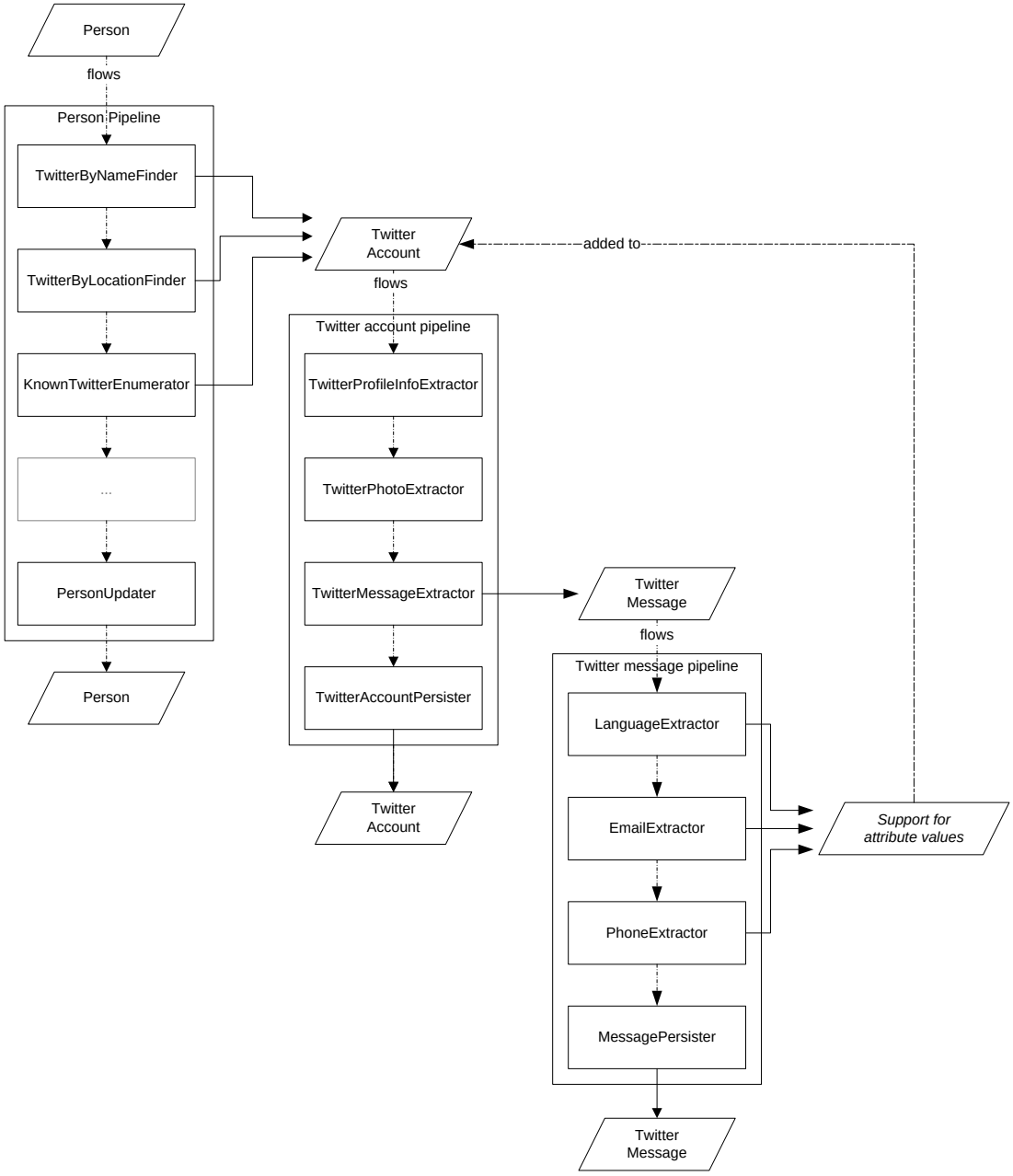


Figure 2.2: Crawler pipeline architecture

a certain possible match is not found in the current crawl, the manifestation should be crawled anyhow. The online manifestation can change over time and some attribute might only be available for a limited period of time. Visiting the manifestation as often as possible increases the chance that it is found. Also, a final sink is added to write any changes to the input person to the database.

The second and third pipeline are responsible for gathering attributes for each candidate match. This starts with a number of sinks that extract information that exists at the account level, like a screen name and location. A sink in this pipeline is also responsible for producing more elementary pieces of data that are part of the online pressence. In this specific case only tweets are produced in the second pipeline. These are then in turn the input for a third pipeline that is responsible for analyzing each tweet and extract for example e-mail addresses, telephone numbers and the language of the message. These attributes are then added to the online manifestation they belong to.

Other data sources are exploited in a similar manner. A number of sinks are added to the main Person-pipeline to provide candidate matches to one or more other pipelines that are responsible for processing all candidates and extracting their features. An other example might be a sink that produces Facebook accounts that possible belong to the given person and enters these into a Facebook pipeline. The Facebook pipeline might in turn extract posts or friends that are entered into a specific pipeline for processing these.

2.3.1.1 Limiting the crawl space

When searching for candidates, the number of possible matches can grow fast. For example, when using a TwitterByName finder it will execute a Google query for every combination of first name, last name and possibly tussenvoegsels [1] and examine the top 25 results for each query. If someone has two first names and tussenvoegsels, this will result in 50 results already. If all of these results are a Twitter account or container, over 50 candidate matches can be added for a person every day. Of course the number of new candidate matches will reduce after two or three runs, but there remains a potential for finding many candidates. Another reason that might happen is including a TwitterByLocationFinder sink for someone who has an address in a city center. Searching Twitter accounts by this location has the potential to yield many new results on a daily basis due to shoppers or people going out and using Twitter.

To prevent the candidate set from getting to large, a form of pruning is used. For each pipeline a Pruner sink is developed, for example the *TwitterAccountPruner*. These pruners remove the candidate match from the candidate set if it receives a very low score from the Matcher, e.g. when it is very unlikely that it is a correct online manifestation, when it has been crawled more than ten times already. This way the number of accounts is bounded by ten times the number of candidates that are discovered every day. Of

[1] Tussenvoegsels are a typical Dutch phenomena. They are used between first and last name, like "Jan van der Sloot". In other languages they are often taken as a part of the last name, like "Jan vanderSloot"

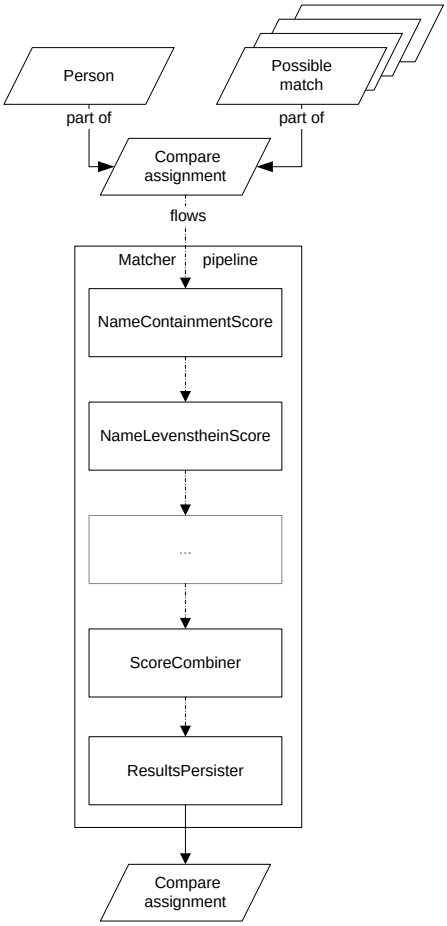


Figure 2.3: Matcher pipeline architecture

course the candidates are retained, so they are not found again by the searching sinks, but just not crawled anymore.

2.3.2 Matching

The actual entity resolution is performed in the matcher pipeline. Again all persons in the input are enumerated, together with all possibly matching instances. The person and the set of possible matches are provided together to the pipeline as a *CompareAssignment*. Different sinks in the matcher pipeline execute a comparison function for every combination of person and possible match contained within the compare assignment. They can add there score to the compare assignment under a specific name so all scores are retained. After all scoring sinks, other sinks can be added to interpreted the individual scores. This can be by taking the normal average of each score, a weighted average or more sophisticated combinations. The different sinks that are available for scoring and

combining scores are described in more detail in Section A.3. Finally, the ResultsPersister is responsible for saving the results back to the database.

2.3.3 Input and output

The input for the IMatcher should be grouped into details about a person. For each person, the following information should be provided;

- A unique number, f.e. the social security number of the person. This number only serves as a unique identifier for this person.
- First name(s)
- Last name
- Tussenvoegsels (if any)

Furthermore, the following can be provided to allow for better searching and resolution:

- One or more addresses the person often resides at. For example living and working address(es). If unknown, less detailed input like countries, cities or streets can be given as well.
- One or more telephone numbers
- One or more e-mail addresses
- One or more aliases used by the person. This can be anything, from a nickname to a online pseudonym or an often used character name for online gaming.

The output of the IMatcher is a result for every person in the input. For every person, two types of results are returned. The first is a statement that "the following online accounts" belong to this person. Secondly, all possibly matching online accounts that were found by the IMatcher, ordered on the probability that they are the correct one, are returned. The latter is mainly for evaluation reasons and for human post-processing.

2.4 Detailed design

This chapter covers more detailed implementation issues. The piplelines that are described in Chapter 2.3 are implemented with a number of sinks that apply, often existing, techniques to perform their tasks. Existing techniques that are not modified, but only applied are discussed in Chapter 5. Newly developed techniques and Techniques that are adopted and modified or extended are discussed in the following sections. A list of all sinks is included in Section 2.4.4.

2.4.1 Evidence based uncertain multi-valued database

Existing theory

Multiple approaches for dealing with imprecise or uncertain databases have been proposed over the last years. There are now both relational and XML databases that support working with uncertain data [22, 35]. In this chapter the notation proposed by Lee [22] is used. This notation accounts for both unkown and uncertain values. The following notation is used to note that an entity P has an attributed named twitter which has two possible values of which the likelyhood is unkown:

$$P.twitter =< \{"@T1", "@T2"\}, 1 > \tag{2.1}$$

To show that both options are equally likely to be true, the following notation is used:

$$P.twitter =< \{"@T1"\}, 0.5 >, < \{"@T2"\}, 0.5\} > \tag{2.2}$$

It is also possible to combine both notations. For example the following represents that there are three possible values: @T1, @T2 and @T3. Furthermore it is know that there is a 50% chance that @T1 is the correct value. Of course it then follows that there is a also a 50% chance that either @T2 or @T3 is the correct value:

$$P.twitter =< \{"@T1"\}, 0.5 >, < \{"@T2", "@T3"\}, 0.5\} > \tag{2.3}$$

The power of this notation is that it allows for combining situations where distribution of probabilities is known and situations where just some possible values are known. Data in this form can be saved into the database and be queried directly to retrieve a set of all possible values. However, it can also be converted by the database on the fly to support retrieving only one value. The database will then return the most probable value.

As powerful as this notation is, it does not allow for denoting the evidence for each value. If, for example, more evidence support for the possibility "@T1" is found, there is no way to calculate how this would influence the existing probabilities. This shortcomming

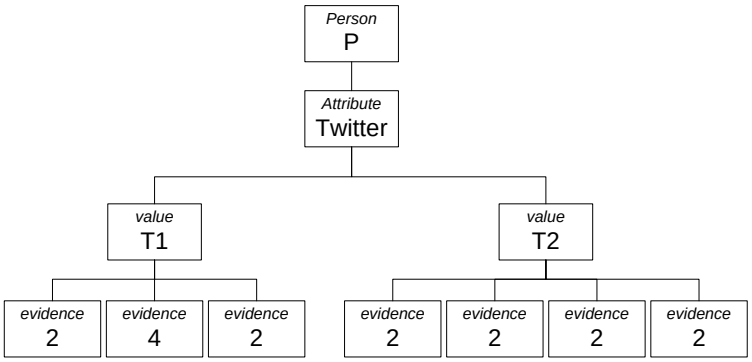


Figure 2.4: Probabilistic values with evidence retained in a form that can be supported by XML

needs to be overcome to allow probabilitic databases to alter probablity distributions over time.

Extending the theory

To make this possible, a new and more elaborate representation of all possibilities and the evidence for each possibility is proposed.

$$P.twitter =< \{"@T1"\}, \{2, 4, 2\} >, < \{"@T2"\}, \{2, 2, 2, 2\} > \tag{2.4}$$

This represents that there two possible values, supported by respectively three and four pieces of evidence. Each piece of evidence is given a weight to allow some pieces of evidence to have more significance than others. In this case Equation (2.4) is equivalent with Equation (2.2), since they both show that both possible values are equally likely. This can be shown by taking the sum of evidence for each option and dividing it by sum of all evidence in (2.4), yielding the same propabilities as in (2.2). However, in Equation (2.4) it is possible to insert a new piece of evidence into the evidence set for a value to adjust the probabilities. For example, inserting a new piece of evidence with weight 16 for the value @T1 would shift the probilities to 75% and 25%.

It is possible to build this new view on probabilistic data into existing databaes. Most databases provide the possiblity to extend their functionality using plugins. However, developing a brand new DBMS (plugin) to support the IMatcher is out of scope for this thesis. For this reason, a Java class has been developed in application space to implement this new level of bookkeeping. In the XML database a specific node structure is used to keep track of all possible values for a node and the evidence for each value. In Figure 2.4 an example is shown to represent the possible values @T1, @T2 with evidence for each value, equivalent to Equation (2.4). When the object P or some of it attributes are needed by the application, the whole object P is transfered to applications pace and represented as an evidence-based uncertain object.

2.4.2 Working with internet sources

The IMatcher has to work intensivly with sources that are available only over the Internet. It interacts, among others, with the Google search or Twitter API. Automatically retrieving information from this type of resources is often refered to as spidering them. Many online sources pose strict limits on the way they can be accessed. For example, Twitter has a strict limit on the number of requests that can be made per hour[1] and Google imposes not only a hard limit on the number of requests but also requires that they are the result of an human action and do not invoked automatically in a quick succession. To work around these limitations paid contracts are offered to increase the maximum number of requests.

One way to get around these kinds of limitations is by using transparant proxies. Transparant proxies are a man-in-the-middle service that take a request for any url and forward it to its intended destination. For the host handling the request it seems that the proxy is doing the request and the limits are imposed on this proxy. When using multiple proxies and rotating among them more requests can be done per hour.

However, handling proxes, sending requests, parsing them and all those other tiny nitty-grity details that need to be handled when spidering a website take a lot of time to implement. To avoid implementing all this, a product under development is used: Neogeo. Neogeo is a project of the University of Twente that can be used to automatically spider complete websites. Although this automated scraping functionality is not used, since no full websites are scraped, substantial parts of this project are used to handle proxies, requesting and parsing HTML pages.

2.4.3 Matching

When, for each person, a set with candidate matches has been gathered, the IMatcher still has to determine which account is the correct one. This is done by calculating a number of scores that each represent the similarity of the person and account from some point of view. All scores that are currently calculated are listed in Table 2.2. This table provides the name of each score, its range and what it reflects. Furthermore, each score can also be null, this means that there was nothing to score.

After running the IMatcher, all these scores will be analyzed to determine which scores can be used best to identify correct Twitter accounts. In other words, for which scorer functions is is there a correlation between the score and the likely that the candidate match is actually the correct Twitter account. However, such an analysis can only be executed after running the experiments (See Section 3.1.3. During the experiments a naive function is used to combine all scores for pruning. This function just takes the average of all available scores. This means that scorer functions that do not provide a score (return null) are not taken into account.

[1] https://dev.twitter.com/docs/rate-limiting

2.4.4 Sinks

Implementing of the pipelines described in Section 2.3 is done using in sinks. Sinks are the atomic building blocks of the IMatcher. Currently there are 22 of these building blocks available. An overview of all sinks is available in Table 2.3, details for all these sinks are provided in Appendix

Table 2.2: List of all scores provided by the Matcher. The approach taken to calculate each metric is discussed in more detail in Appendix A.

Name	Range	Description
country	0 .. 1	Describes how much of the found geotags are in the same country as at least one of the addresses of the subject
city	0 .. 1	Describes how much of the found geotags are in the same city as at least one of the addresses of the subject
house-number	0 .. 1	Describes how much of the found geotags are in the same street as at least one of the addresses of the subject
street	0 .. 1	Describes how much of the found geotags are at exactly the same address as at least one of the addresses of the subject
language	0 .. 1	Describes how much the language of all found tweets are in a language that is known for the person
email	0, 1	Describes how much of the e-mail addresses in all tweets match one of the given e-mail addresses
distance	0 .. 1	Describes how far, on average, each geotag was from the nearest known address. For each found geotag, the distance to the nearest address is calculated in km. The average of all these distances is then divided by 20.000 km, which is roughly the furthest two points on earth can be from each other.
distance-close	0 .. 1	Describes how far, near geotags were from the nearest known address. This is calculated analogue to *distance*, but with a divider of 0.5 km
name-containment	0 .. 1	Describes the similarity of the name of the provided account and person. This is done by exploding the name of the person on whitespace and determining the how many of the debris can be found as a substring of the account name. This number is divided by the total number of debris.
name-levenshtein	0 .. 1	Describes the similarity of the name of the provided account and person. This is done by calculating the edit distance for all permutations of both names after exploding them on whitespace. The lowest edit distance is divided by the length of the longest input name and than returned.

Table 2.3: List of all sinks in the IMatcher. Their I/O type and a description are given. If an extra type is generated as side-output -and thus input for another pipeline- it is provided as well.

Name	I/O type	Side-output	Description
KnownTwitterIterator	Person	TwitterAccount	Enumerates all twitter accounts currently in the candidate set, however prunes very unlikely candidates
TwitterByGoogle	Person	TwitterAccount	Searches for new candidates by name of the person
TwitterByLocation	Person	TwitterAccount	Searches for new candidates by known addresses
PersonPersister	Person	-	Saves changes to the person representation to the database
TwitterGeneralInfoExtractor	TwitterAccount		Extracts a free form location, url, account name, followers count, e-mail adres and telephone number from a Twitter profile.
TwitterPictureExtractor	TwitterAccount	ProfilePicture	Extracts the Twitter profile picture
TwitterMessageExtractor	TwitterAccount	TwitterMessage	Retrieves and enumerates all twitter messages and enters them into the appropiate pipeline
TwitterAccountPersister	TwitterAccount	-	Saves changes the accounts representation to the database
TwitterMessageGeoExtractor	TwitterMessage	-	Extract geotags from twitter messages
TwitterMessagePersister	TwitterMessage	-	Saves all twitter messages to the database
MessageEmailPhoneExtracter	Message	-	Extracts all e-mail addresses and telephone numbers from the message
MessageLanguageExtracter	Message	-	Determines the language of the message
TwitterPossibilitiesAdder	TwitterMatching	-	Adds all accounts in the candidate set to the TwitterMatching compare assignment
TwitterEmailMatcher	TwitterMatching	-	Calculates a match score based on e-mail for each candidate match
TwitterTelephoneMatcher	TwitterMatching	-	Calculates a match score based on telephone number for each candidate match
TwitterGPSMatcher	TwitterMatching	-	Calculates the distance and distance-close score
TwitterLanguageMatcher	TwitterMatching	-	Calculates the language score

TwitterLocationMatcher	TwitterMatching	-	Calculates the country, city, street and house-number scores
TwitterGPSMatcher	TwitterMatching	-	Calculates the distance and distance-close score
TwitterNameContainmentScore	TwitterMatching	-	Calculates the name-containment score
TwitterNameLevenshteinScore	TwitterMatching	-	Calculates the name-levensthtein-score
ScoresCummulator	TwitterMatching	-	Calculates a new score average, which is the average of all non-null scores
TwitterMatchingPinter	TwitterMatching	-	Prints all scores

Chapter 3

Experiments

To assess the performance of the developed prototype under different conditions two experiments have been conducted. Each of the experiments is discussed in its own section. The experiments in Section 3.1 performed on subjects that provided their details using full informed consent. The goal of this experiment was to determine the accuracy of the IMatcher and to investigate how the accuracy is influenced by the absence of specific information. This is possible, since there is a ground truth available: for each person it is known if they had a Twitter account and, if any, what their user name is. The experiments from Section 3.2 were run using the details of 85 persons that were owner or manager of an job agency that is part of a current risk analysis. For these subjects no ground truth is available so they are not taken into account when assessing the quality of the IMatcher, the results are only provided back to the ISZW.

3.1 Experiment 1

The goal of this first experiment is to measure the performance of the IMatcher. In the research questions posed in the Introduction two variables that possibly influence the accuracy of the IMatcher are discussed. These variables are the completeness of input data and the duration of the experiment. By deliberately removing some of the known information about a subject from an installation of the IMatcher, it is investigated how much the accuracy decreases when that information would not be known. This

shows the relation between the availability of that specific information and accuracy. By running the IMatcher a number of times and analyzing the results between each run, it is investigated if the accuracy of the results changes over time.

Accuracy is measured at the run level using the average candidate *setsize* and the *inclusion* of correct Twitter accounts. At the subject level recall and precision are measured after running a classification algorithm is run to classify candidate accounts as correct or incorrect. All measurements of *setsize* are before pruning.

3.1.1 Method

This first experiment was performed using four installations of the IMatcher, all running in complete isolation. These four installations made up four variants of the experiment that were run. Each variation was initialized with a different level of completeness of the input data. The first variation was initialized with all information provided by the subjects, being: their full name, all addresses provided, e-mail address, telephone number and the language spoken. In the second variation all last names were left out of the input information. In the third variation all addresses were left out. Finally, in the fourth variation all e-mail addresses and telephone numbers were left out. Thus, in each variation only one type of information was left out.

These four variations were chosen based for a number of reasons. The first variation is a baseline measurement to establish the accuracy if all available information is used. The second variation was based on a number of observations in related work about the fact that many people use their real name on social network sites. For example, Veldman [36], Perito et al. [30], Motoyama et al. [27] and [24] all concluded that the full name of a subject is not only often found on a social network profile but is also very reliable and is a very suitable attribute to find and identify users. The third variation was left out to explore if including location information in the search has added value. Currently, the IMatcher tries to find subjects using a series of Google queries and by gathering Tweets in a circle about known addresses. It seemed logical to explore whether using this search by location has added value. Finally, e-mail address and telephone number were left out since other researchers reported that they also were valuable attributes when identifying users.

3.1.1.1 Subject selection

For this experiment, subjects were self-selected by signing up under full informed consent. In total 22 subjects signed up for the experiment. Possible subjects were informed about the experiment and asked to sign up in person and some acquaintances posted a Twitter message to attract more subjects after signing up. This is a convenience sample consisting of a few acquaintances and mostly acquaintances of acquaintances of the author. Of the subjects 16 were male and 6 were female. Of these subjects, 12 owned a Twitter account.

A distribution is shown in Table 3.1.

They were informed that the goal of the experiment was to identify their Twitter account, if they had one. They were also told that for this exploratory type of research and data mining techniques in general, it is hard to predict if there will be any results, and if there will be, what form they will take. All this was done on a website that was developed specifically to inform candidate subjects and allow them to sign up.

Also, the selection of subjects and and how they were informed and threated during the experiments was guided by concept guidelines from the Faculties Ethical Committee. For completeness the original proposal that was sent to the Ethical Committee (in the making) to get permission for the experiment is attached as Appendix B. In the end, no response was received from the Ethical Committee with a go ahead or abort. However, their guidelines have been followed throughout the experiments.

Table 3.1: Details about subjects sex and whether they have a Twitter Account or not

Sex	**Has Twitter**		**Has no Twitter**	
Male	62,5%	(N=10)	37,5%	(N=6)
Female	33,3%	(N=2)	66,7%	(N=4)

3.1.1.2 Collected variables

All subjects were asked to provide their full name, a number of physical addresses they often visit, e-mail address, telephone number and Twitter account. All subjects, except one, gave all the requested information. That one subject only provided his initial instead of his first name. All information, except the correct twitter account, was provided to the IMatcher for the experiments. The correct Twitter account was used after the experiments to determine for every candidate account if it actually belonged to the subject or not. The gathered information was not used for anything else.

After each run the IMatcher computed a list of candidate matches for every person and scored their similarity. For each run, this resulted in a CSV with one row per candidate match. Every row had the variation number in it, the number of the subject, the possibly matching account, all scores provided by the comparison functions and two depended variables *hasTwitter* (0 or 1) and *correctTwitter* (0 or 1). The first was directly related to the subject, describing if the subject has a Twitter account (1). The second described if the given candidate account actually belonged to the subject.

The advantage of putting the results in this form is that it makes them usable for classification. Classification is a machine learning problem that is concerned with determining in which category a record belongs based on its properties. In this case, the scores are the properties and the fields *hasTwitter* and *correctTwitter* are the categories. After the experiment all results will be analyzed to see if a classification algorithm can

reliably determine if a candidate is correct given only the scores. See Section 3.1.3 for this analysis.

From these raw, row-by-row, results other variables can be deduces as well. These variables are at the run-level, instead of the subject-level and are *inclusion* and *setsize*. Inclusion is the number of correct twitter accounts that are included in the candidate set. *Setsize* is the average size of the candidate set per subject.

3.1.1.3 Procedure

Between 2 April 2013 and 12 May 2013, the IMatcher was run a total of 46 times for experiment 1. All these runs are listed in Table 3.2. For each run the start and end time is given.

As can be seen in the duration of the runs, there was a significant change in the IMatcher around 3 May. Preliminary investigation of the results at this time showed that the gathered Twitter accounts using heuristic searching did not always contain the correct account, even if it was quite obvious like twitter.com/{first name}{last name}. For this reason, at 3 May the IMatcher was altered to explore no longer the top 8 of the Google results, but the top 20. Also, for each query "Twitter X Y" a new query "X Y site:twitter.com" was added. This was done to increase the chance that the correct account would be included in the heuristic search at the cost of multiplying run times by a factor of 10.

3.1.2 Results

After each run the results up to, and including, that run were gathered. Figure 3.1 shows for every variation of experiment 1 the inclusion: the number of correct Twitter accounts that were found up to and including each run. In total 12 of the subjects had a Twitter account and it can be seen that ultimately 11 of these accounts were found in the last run of variation 1. Before the changes on 3 May only 7 accounts were found in the last run. Figure 3.2 shows for every run the size of the candidate set after that run. It can be seen that the change on 3 May had increased the candidate set by a factor of roughly 10.

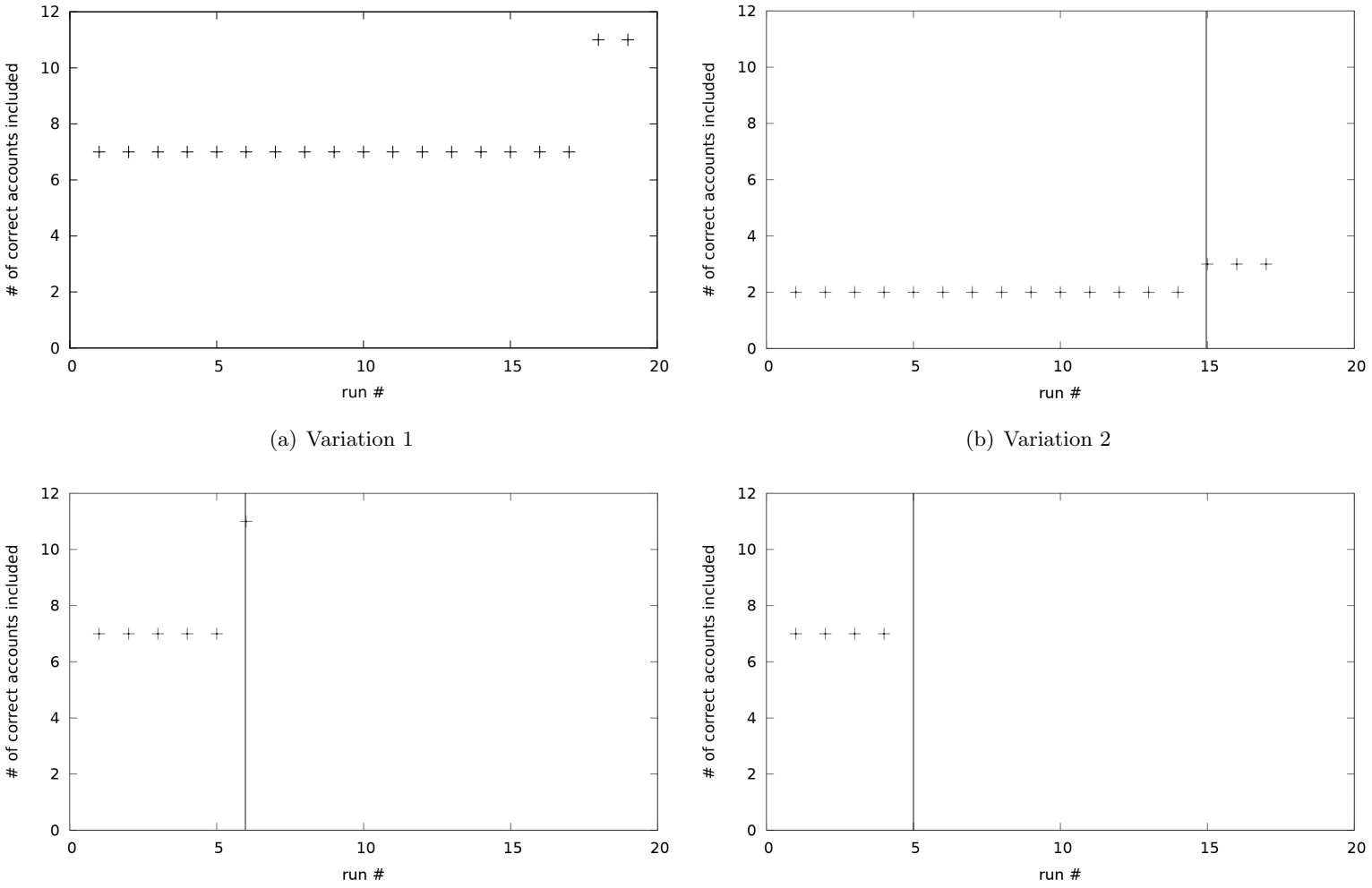
There were in total 22 accounts appeared in 4 candidate sets or more. The top 10 was dominated by accounts like Flickr, SlideShare, 112Twente, peekyou and some popular individuals. There were in total 100 accounts that appeared in 3 candidate sets and 525 accounts that appeared in 2 candidate sets.

As can be seen in Figure 3.1. Even in the most successful variation, variation 1, for one person the Twitter account was not included in the candidate set. The owner of this account did not include his last name in either the username or provided name. Also, this account was marked as protected, which means that its tweets are not publicly available. Interestingly, for this subject and an acquaintance of her, an extra account that they

Table 3.2: List of all runs with the IMatcher for experiment 1. De first column (V) is the variation, the second column (#) the number of the run.

V	#	Start	End	V	#	Start	End
1	1	2 April 12.40	2 April 17.40	2	5	10 April 14.19	9 April 16.24
1	2	3 April 18.26	5 April 21.15	2	6	11 April 18.07	10 April 20.53
1	3	4 April 18.12	4 April 20.30	2	7	13 April 13.59	11 April 17.21
1	4	8 April 15.45	8 April 18.58	2	8	14 April 12.10	13 April 14.20
1	5	9 April 12.18	9 April 15.22	2	9	15 April 17.21	14 April 20.14
1	6	10 April 11.18	10 April 14.16	2	10	17 April 16.26	15 April 19.48
1	7	11 April 15.00	11 April 18.03	2	11	19 April 15.42	17 April 19.55
1	8	13 April 17.23	13 April 22.14	2	12	20 April 17.38	20 April 20.52
1	9	14 April 14.42	14 April 18.11	2	13	15 April 13.16	15 April 6.57
1	10	15 April 13.16	15 April 6.57	2	14	17 April 11.09	17 April 15.35
1	11	17 April 11.09	17 April 15.35	2	15	unknown	unknown
1	12	19 April 09.20	unknown	2	16	unknown	unknown
1	13	20 April 16.55	20 April 21.28	2	17	12 May 10.12	13 May 19.12
1	14	21 April 14.00	21 April 17.26	3	1	20 April 21.66	20 April 23.29
1	15	22 April 12.03	22 April 18.02	3	2	21 April 17.42	17 April 18.34
1	16	unknown	unknown	3	3	22 April 16.46	22 April 17.53
1	16	unknown	unknown	3	4	unknown	unknown
1	18	5 May 23.39	8 May 02:32	3	5	unknown	unknown
1	19	10 May 09.28	12 May 02.22	3	6	6 May 19.37	8 May 18.21
2	1	3 April 10.00	2 April 12.45	4	1	20 April 22.26	21 April 03.22
2	2	4 April 11.14	5 April 13.20	4	2	21 April 19.18	21 April 21.25
2	3	8 April 22.25	4 April 00.30	4	3	22 April 21. 07	unknown
2	4	9 April 15.29	8 April 18.12	4	4	unknown	unknown

shared was discovered in the candidate sets.



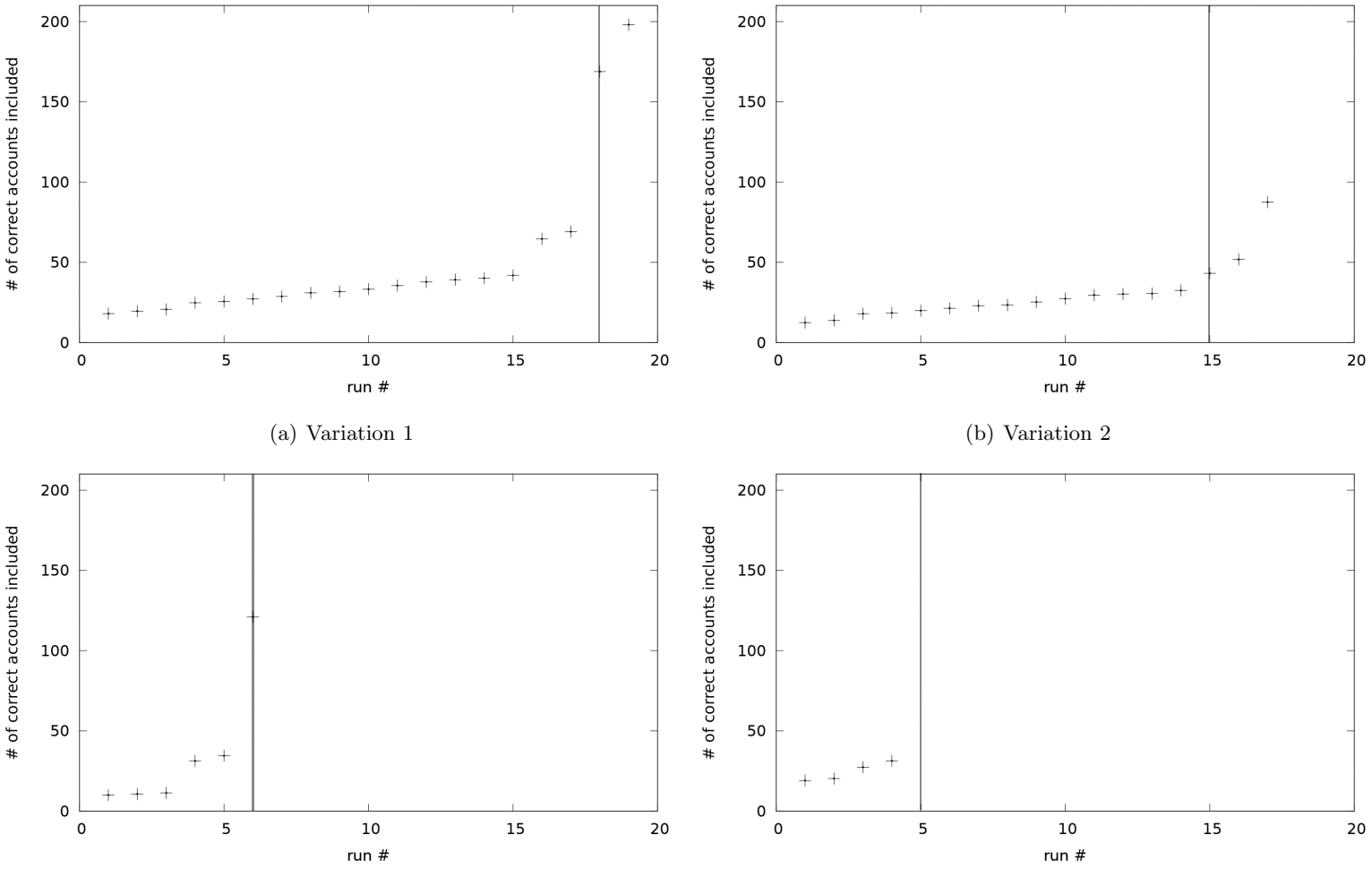
(a) Variation 1

(b) Variation 2

(c) Variation 3

(d) Variation 4

Figure 3.1: The number of correct Twitter accounts included in the candidate set for each variation of experiment 1, after each run. In total 12 of the subjects had a Twitter account. The grey line marks the first run after the 3 May update. After this update the top 20 instead of the top 8 results of each Google query were explored. Just before this update the IMatcher was run a little less frequent, which explains the little jumps in measurements in the few runs before the 3 May update.



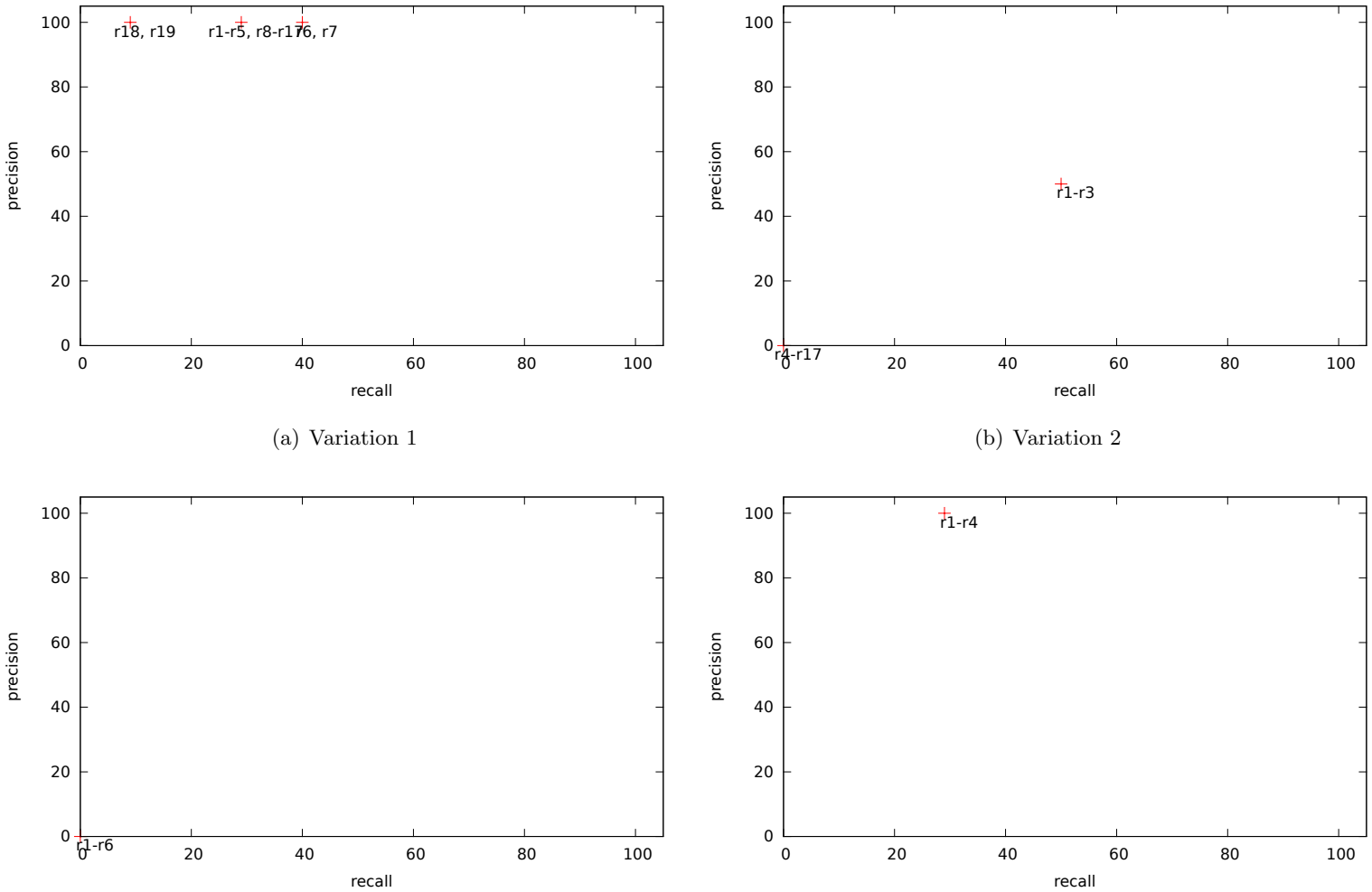
(a) Variation 1

(b) Variation 2

(c) Variation 3

(d) Variation 4

Figure 3.2: The average size of the candidate sets for each variation of experiment 1, after each run. The grey line marks the first run after the 3 May update. After this update the top 20 instead of the top 8 results of each Google query were explored. Just before this update the IMatcher was run a little less frequent, which explains the little jumps in measurements in the few runs before the 3 May update.



(a) Variation 1

(b) Variation 2

(c) Variation 3

(d) Variation 4

Figure 3.3: The recall and precision of machine learning on the results of a run.

Input	Imatcher results		Known Truth	
Subject	Possible match	Similarity scores	hasTwitter	correctTwitter
Barack Obama	BarackObama	{0.9; 0.8; null; 0.3; ....; 0.21}	1	1

Figure 3.4: Input for classification. Results from the IMatcher are merged with known truths to train a classifier. Please note that although in this specific example the subject and Twitter account have an almost similar name, this does not hold for all cases.

3.1.3 Analysis

Now the IMatcher has produced a list of candidate matches and similarity scores for every subject it is time to determine if a classifier can reliably determine if a candidate match is actually correct or not. To do this, the results of the IMatcher are augmented with known truths about whether a subject has a twitter account and, if the provided candidate is correct. (It follows that if a person has no Twitter account, no candidate at all is correct.) An example is shown in Figure 3.4. This Figure shows that for the subject Barack Obama a possible candidate has been found, the scorer functions have determined scores that describe the similarity between the person Barack Obama and the Twitter account twitter.com/BarackObama, that it is known that Barack Obama indeed has a Twitter account and that the given candidate match is the correct one. In total, the 46 runs of the IMatcher generated a little over 32.000 of these rows, grouped into 46 datasets, each representing the cumulative results after a run.

On these datasets a classifier was trained and ran to determine how well a such an algorithm can predict whether a subject has a twitter account and, if so, if the given one is the correct one using only the IMatcher results. For this analysis the scores for each combination of subject and candidate account were used as independent observations and the variables *hasTwitter* and *correctTwitter* were used as category (dependent) variables. In SPSS three different classification algorithms were tested on the results of run 1 of variation 1. This showed that both decision trees and neural networks classified every candidate account as being not correct. Only a binary logistic regression was able to correctly classify some candidate accounts as correct. This is most likely due to the unbalance in the datasets. Since only between 1% and 5% of the rows in the datasets concerned a correct twitter account, classification algorithms were very likely to just call everything incorrect, resulting in a very high overall correctness but identifying very little correct accounts.

Since binary regression produced the best results, this classification algorithm was run for all 46 result sets. The results of these runs are gathered in Figure 3.3. This figure shows per variation the relation between precision and recall for the classification. It can be see that the precision and recall does rarely change between runs. In run 1, the results were improved slightly for run 16 and 17, but dropped down much more for run 18 and 19. These two runs were after the update of 3 May and resulted in even more

unbalanced datasets. It is interesting that precision is very high. Every account for which the classifier claims that it is correct, the classifier is correct. However, it is only able to determine a very low percentage (8% - 40%). For variation 2, the initial results were better, however they dropped down to a recall of 0% for runs 4 to 17. This means that precision is formally not defined, and was therefore noted as 0%. The results of run 3 show that no account could be identified correctly, again resulting in a precision and recall of 0%. Finally, run 4 has stable results with again a precision of 100% and recall of only 33% percent.

After this analysis, classification was performed again including the variable *hasTwitter* as a independent variable. This was done to see if knowing that a person has Twitter, increases the chance of identifying it correctly. However, adding this variable to the list of independent variables did not influence the results significantly. In all cases, recall either remained at 33% or increased slightly to 42%. However, precision dropped down to around 40% to 60% percent.

3.1.4 Conclusions

The experiment described above was performed to investigate the existence of two relations. The first is a relation between completeness of the IMatcher input and the accuracy of the results. The second is a relation between the number of runs and the accuracy of the results. From the performed experiment a number of conclusions can be drawn.

First, *inclusion* is very high for variation 1. Especially after the update of 3 May, inclusion rises to 91% (11 out of 12). This shows that it is possible to find a person's Twitter account using the IMatcher. However, this comes at a price. The average *setsize* also grew with a factor of 9 after the update of 3 May. Quickly pruning false positives from the candidate set will be important to prevent an explosion of candidates that have to be crawled. However, from this it can be concluded that it is viable to retrieve a person's correct Twitter account using a set of heuristic searches, together with a lot of non-correct accounts.

Secondly, variation 2 shows that leaving out the first name of a subject reduces *inclusion* significantly, to 17%, but does not influence the average *setsize*.

Thirdly, variation 3 shows that leaving out addresses of the search does not influence *inclusion*. However, it does have the benefit that it reduces the average *setsize* significantly to approximation 1/3 of the average *setsize* of variation 1.

Finally, variation 4 shows that leaving out the e-mail address and telephone number of the subjects has no influence on either the *inclusion* or the average *setsize*.

Finally, the regression analysis showed that using only the scores that the IMatcher currently provides cannot be used to reliably determine if a candidate account is actually the correct account. The only observation that can be made is that leaving out the

Table 3.3: List of all runs with the IMatcher for experiment 2

#	Start	End
1	19 April 23.55	20 April 17.33
2	20 April 23.57	21 April 11.51
3	21 April 21.29	22 April 08.29
4	22 April 21.07	unknown
5	5 May 23.39	11 May 22.55

addresses of a subject reduces the accuracy of classification. This is most likely due to the fact that incorporating addresses (and thus activating the scorers based on geotags of the Tweets) does have a positive influence on the classification results.

3.2 Experiment 2

This experiment was commissioned by the ISZW to see if they could use the results in their ongoing risk analysis'. A total of 85 subjects were provided by the ISZW, without their consent as part of ongoing investigation. Due to this procedure there is no ground truth available about the subjects. This means that it is not known which subjects have a Twitter account and for those who have it is not know what is their Twitter account. From a research perspective this experiment only provides more insight into the scalability of the IMatcher.

Since there is no way to determine the accuracy of the results, there were no variations within this experiment. Only one installation of the IMatcher was set up and all available information about the subjects was used as input. This information consisted of their full name and working address. No home addresses, e-mail addresses or telephone numbers were provided.

With this installation of the IMatcher only 5 runs were ran, mostly due to the length of the final run, which took almost 6 days. All runs are listed in Table 3.3.

The result of this experiment was the average *setsize* for each run, they are shown in Figure 3.5.

All results, meaning the candidate set for each subject and the scores provided by each scorer function were also handed over to the ISZW.

3.2.1 Conclusions

This experiment provided no important conclusions, except for the observation that the average *setsize* is comparable to those of Experiment 1. The *setsize* is a little lower than that of Experiment 1, Variation 1 and higher than the *setsize* of Experiment 1, Variation

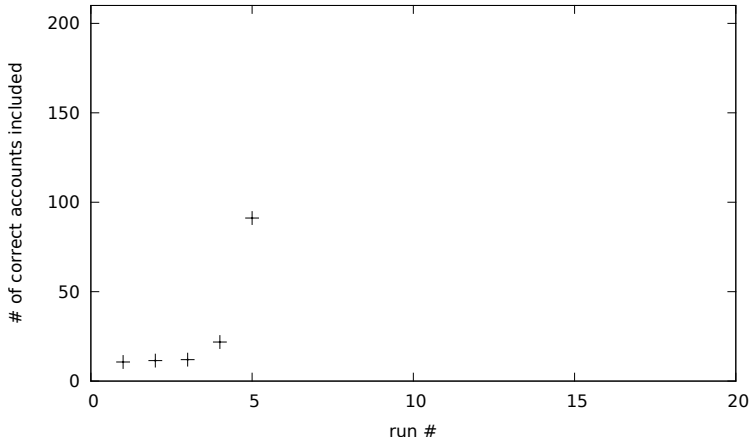


Figure 3.5: The average size of the candidate sets for experiment 2, after each run. The grey line marks the first run after the 3 May update. After this update the top 20 instead of the top 8 results of each Google query were explored.

3 which is due to the fact that only one address was known per subject. This indicates that the context of the subjects sample does not influences the average *setsize*. Also, the same increase, by a factor of 9, is observed after the update of 3 May.

Also, the duration of the runs seem to scale more or less linear. Although there was a factor four increase in the number of subjects, there was only a factor 3 increase in the run duration. This is most likely due to the the fact that subjects in experiment 1 had, on average, more addresses provided than the subjects in experiment 2. This increases the number of queries and thus the run time.

3.3 Experimental conditions

All experiments were conducted on a laptop equipped with an Intel Core i3 330M processor with 6GB of RAM and a 5200rpm hard disk drive, running Ubuntu Linux 12.04 and BaseX 7.3. During the experiment, 8 external transparent proxies were used to submit all request to the various web services that were used. All data was written to an encrypted portion of the filesystem using EncFS. This type of encryption takes a surprising amount of CPU power and limits disk access caching so there is a lot of disk I/O and one CPU core is continuously at 100%.

Chapter 4

Discussion

4.1 Known limitations

The current prototype has many shortcomings. Although some of them are fairly easy to address, they often have not been, to make sure all measurements were taken under the same conditions.

First, because they either live or work together, there are a number of subjects that have provided the same address. Due to a design error, all candidates which were discovered to tweet in a circle around these addresses were added to the candidate set for only one of these subjects. This was almost always the candidate set of the first subject encountered. This was the case for a total of four subjects, who shared a total of 2 addresses.

Secondly, some people have their Twitter account marked as protected. This means, that tweets cannot be read by anyone except for those explicitly allowed. This should be, but was not, taken into account while performing the identity matching. Also, it should be considered that the contents of these accounts rarely change for those who do not have access. Since the Google crawler does not have this access it does not see change on the account and will therefore rank it much lower in the search results. (To be more precise, Google will put recently changed documents higher in the rankings.) For this reason, protected accounts might be overlooked in the experiments. In total, two subjects had a protected account.

4.2 Future research

There are many research directions that still need to be explored.

on queries
First, the 3 May update showed that *inclusion* could be boosted from 58% up to 92% by exploring the top 20 results for each heuristic query, instead of the top 8 and adding a query variation. This shows that more research is needed to explore which exact queries yield the best results and how many results for each query need to be considered. It might also be interesting to explore search broadening or iterative deepening as a strategy. For the first few runs only the top 10 results are explored and if this yields no clear result, the top 20 is explored, then the top 30, until a certain threshold is reached below which no results should be expected.

Also, completely different types of queries should be considered. For example, Google has an API which allows search for images. This works very well trying to find all occurrences of the same image on the Internet. Using a technique like this, other online manifestations of a person could be found after discovery of the first one. For example, after finding a Twitter account the images linked to that account might also appear on a specific Facebook profile. It might be likely then, that this profile belongs to the same person. Another example might be topic-related. For example, when investigating a group of subjects that are all engaged in asbestos removal, following specific Twitter hash tags like #asbestos and/or #removal feeds might yield candidate matches that can be added to more then one candidate set.

on identity matching
More research is needed to establish a better matching function. The results of Experiment 1 show that, although *inclusion* is very high, it is still impossible to identify matching Twitter accounts due to a lack of discriminating attributes. To improve these results more features need to be extracted from the Twitter accounts and better scorer functions need to be developed. For example, gender and age extraction might return two attributes that can be used to improve the identity matching results. On the other hand, image similarity can prove very viable to link multiple online manifestations together. There are many directions to explore and most of them will prove to be projects in them selves.

The location provided by subjects was not only used to search on, but also by multiple scorer functions. Unfortunately for the IMatcher, not every tweet had a geotag attached. The result of this was, that for most Twitter accounts no locations could be extracted. This then resulted in no scores provided based on location by the different scores, which made identity matching more difficult. Future work should also focus on extracting locations from tweets or accounts themselves. Many accounts have a less formal description of their location like, "somewhere in Twente" or "A small place in Groningen". Also, many Tweets contain some hint about their location, like "Going to Enschede tomorrow" or "Bummer, again a delay at Utrecht" These less formal locations should also be extracted to provide much more accurate scores and improve identity matching results.

Future work should also investigate the relation between time and location. Although many people might go to a specific gym, there is only a small group that goes to that gym on specific days, on specific times. Harvesting this combination of location and time might prove to enhance identity matching. This is supported by related work (See Chapter 5).

beyond Twitter
Currently the IMatcher is written as a framework that can be used to explore more than one social network site [SNS]. However, it still requires a specific wrapper per SNS. In future research more wrappers can be developed to find other online manifestations on, for example, Facebook, Marktplaats or Ebay. Also, more free-from sources like personal blogs can be included in the search. This would require developing a retrieval component that is no longer tailored to a specific source, but can handle any type of web page.

4.3 Scalability

As was presented in Chapter 3, runs of the IMatcher take up a lot of time. This makes that the current prototype is not very scalable. There are four main reasons for this.

First of all, the newly developed look at probabilistic databases is not implemented in the database itself, but at application level. The consequence of this is that all probabilistic data surrounding an entity has to be transfered from the database to application space (and back) every time only one attribute of that entity is needed. For example, the persisted version of a full Twitter account, with all its attributes and evidence supporting each value is transfered from the database to the application and back every run. If the probabilistic view would have been included at the database level, only the possible values for an attribute could have been retrieved and evidence added. This would result in much less disk I/O, which would be much faster. Implementing the newly developed view on probabilistic data at the database level can remove this bottleneck.

Second, in the current setup, the file system was encrypted. This reduced read and write speed for the file system to 14,5 MB/s at full CPU load. Normally, the disk used would perform around 68 MB/s. Therefore, the encryption introduced another bottleneck. Also, this 14,5 MB/s is at maximum usage of the CPU. If the CPU would be doing other things as well, like running another IMatcher instance, the I/O speed might be reduced even more. Using a server with more CPU power might counter this. Ideally a system without encryption is used. However, for these experiments this was not possible since the experiments were run on a laptop that was often moved and therefore not physically secured.

Third, the IMatcher has a mechanism for pruning built in, however it barely reduced run times. Pruning means that accounts in the candidate set that are very unlikely to be correct are excluded in the crawls after some time. During this experiment pruning was done if the average over all scores was below a certain threshold. Since the best

way to determine a final score from all sub scores could only be determined after the experiments, his threshold was very low. This was done on purpose to guarantee that no correct accounts were pruned. Since analysis was done only afterwards, this threshold had to be low enough to make sure that every correct account got through. As a consequence, fewer accounts were pruned during runs than might have been possible. Longer running experiments will most likely benefit more from pruning.

Fourth, despite all this, the biggest slow down was -surprisingly- all the calls to external web services. There are two ways these external services delay execution. First, most services have a maximum number of calls they will service, per hour. For example, unauthorized Twitter calls can be done at most 150 times per hour. This limit can be increased by using multiple outgoing IP's and rotate over them. This is done by using the Neogeo component and transparent proxies. This was known in advance and IMatcher instances were throttled -if needed- to honor these limits. However, after adding geotag extraction to the IMatcher, the number of web service calls exploded, due to the fact that the Google API was used to translate these latitude/longitude GPS locations into addresses. In the end, this was the most delaying factor: All the small delays incurred by calling external web services. Not only the call itself, which had to be routed through a proxy, sometimes also a slower processing due to throttling by the service. Or even worse, a complete denial of service which means the request had to be send again using another proxy. This can easily be counted by multiplying the IMatcher pipeline and have them work in parallel.

The influence of all these delays is stronger since execution of the whole IMatcher is sequential (due to its pipeline architecture). This means that, while there is a long running web service call being done the whole system is waiting. And that when a large record is being persisted to the database, the Internet connection is idle. Running operations in parallel can significantly improve runtime by using multiple resources of the system at the same time and not sequentially.

All these factors contributed to very long running executions of the IMatcher, up to almost 6 days for the longest run. However, addressing the points mentioned above might reduce this considerably.

Chapter 5

Related work

5.1 Uncertain databases

Recently the idea of certain, and always correctly filled, databases has received quite some criticism. Researchers claim that there are many reasons for which values in a database are not as definite as they appear. For example, a scheduled meaning for which the location is blank can have different meanings. For example, the location is not decided upon yet, or the location was not know to whomever entered the meeting in the database. Or it is not decided yet if the meeting will take place at location A or location B, therefore making it impossible to enter it into the database. Bosc and Pivert [7] identify two types of uncertainty that can arise when working with data: attribute uncertainty and existential uncertainty. The first relates to situations where the value of an attribute is uncertain, like before. The second relates to situations where it is unknown if an attribute even exists.

A well accepted approach to uncertain database (or imprecise, probabilistic or incomplete databases [22]) is that the possible worlds model. In essence this model states that if it is impossible to capture the real state of the world in a database, the only thing left to do is enumerate all possibilities that are consistent with is known about the real world. Often a probability is attached to every possible state of the world to describe how likely it is. A query will then return all possible answers, ranked on the probability that they are *the* result. A known trade-off of this model is the performance penalty [7].

A syntax for describing uncertain values is proposed by Lee[22], which is described and extended in Section 2.4.1.

The concept of uncertainty has also been taken to XML. For example, van Keulen et al. [35] propose probabilistic nodes that can be used to encode the possible worlds model in XML. A more theoretical approach to uncertainty in XML documents is taken by Abiteboul et al [1] who describes five types of nodes that can be used to encode uncertainty. He also shows that all proposed notations for probabilistic XML that provide a subset of these five nodes can automatically be translated into each other: they are equivalent.

Although there are currently some relational databases that provide the possibility to define tables with uncertain values, there are no production-ready XML databases that support uncertainty.

The current possible worlds model does not allow for incorporating new evidence for one possible value and shifting the probability distribution between all possibilities. In this thesis an extension to the existing possible worlds model is proposed that does allow for adding evidence support to values and thereby increasing and/or decreasing the probability of specific values.

5.2 Entity resolution

This section has also appeared in another paper [5] and has only been slightly altered.

Entity resolution, also known as merge-purge, is the proces of finding records in multiple data sources that describe the same entity. Entity resolution is characterized by the fact that it works on multiple data sources, opposed to deplucation or record-linkage techniques [8]. Any approach to entity resolution consists of three important parts: a comparison function, a merge function and an algorithm that governs the application of those two functions [6]. With regards to the time necessary to perform entity resolution Benjelloun et al [6] note that often the most computation time is needed for executing of the compare function, so much research has been done to investigate different application algorithms and optimizing them. In their survey Brizan and Tansul [8] differentiate between four types of entity resolution algorithms: naive, sliding window, bucketing and hierarchical.

The naive approach consists of simply comparing every entity in one set to every entity in the other set. The advantage of this approach is that it every possible pair is processed by the matching function and matching pairs -according to the matching function- are always discovered. The disadvantage is that both the worst-case and average-case complexity linear is in the size of both documents: O(n * m).

Sliding window approaches work in two steps [15]. First all records are sorted on some attribute of the records and a window of size w is opened at the position of the first

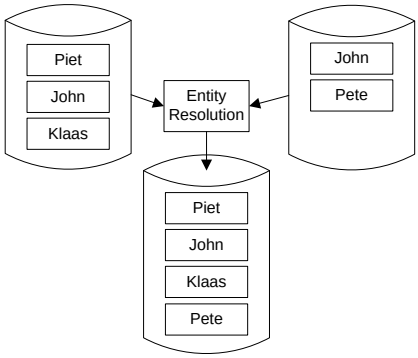


Figure 5.1: Entity resolution: Merging multiple sources while preventing duplicates

record. After comparing all records within the window, the window is moved 1 position and again all records within the window are compared to each other. This step repeats until all records are processed. The advantage of this approach is that comparison is linear in the size of both datasets. However, records need to be sorted first, so the overall complexity is O(n lg n). The disadvantage is that some matching records might be missed, since (fuzzy) sorting does not guarantee that matching records are always placed less then w positions apart. To counter this, sometimes multiple attributes are used for sorting and the whole procedure is repeated once or twice. An example is the approach proposed by Whang et al [39] that used multiple attributes in an iterative manner to both increase accuracy as well as speed.

In a bucketing approach an extra mapping function is introduced that maps all entities on to buckets. The approach compares to hashing, with the exception that when hashing two almost identical entities the resulting hashes are ideally not close together whereas putting two almost similar entities into the mapping function they should end up in the same bucket. After this process, the naive approach is used to identify matching entities per bucket. The worst-case complexity for this approach is dominated by the single-pass through the bucket function and therefore O(n) [8]. Bucketing has the same disadvantage as sliding window approaches, it can miss possible matches if matching records, for what reason, do not end up in the same bucket.

In their paper Brizan and Tansul [8] state the belief that grouping approaches, like sliding windows and bucketing, do not always work. Not every type of record can be hashed, grouped or sorted in such a way that it is at least close to other records that describe the same real-world object. Hierarchical data structures provide a special case for entity resolution where the form of the data is used to speed up entity resolution without the risk of loosing accuracy.

There are many approaches to setting up similarity functions. Often the function used is tailored to the datasets on which entity resolution is performed. This can be a simple a function as the levenshtein distance (See section 5.4.1) or an adaptable approach. A more recent approach is to use relationships with other entities. An often used dataset is

that of co-authorships, however the first paper on this approach by Ananthakrishna et al [3] only showed detailed experimentation on a customers database. They started with a dataset that was known to be clean and introduced duplicates on purpose to verify their approach and show that relationships are a viable approach to similarity detection. A similar approach was also used by Veldman [36].

In this thesis, a new approach to entity resolution is introduced. First an heuristic query is executed to build a set of candidate matches. Only on the original entity and the candidates the comparison function is executed. The taken approach is therefore a grouping approach. This is done since it is not feasible to compare a person to every Twitter account in the world. The difference with a bucketing approach is that not all entities are processed, but that external parties are queried for a set of candidates. This means that it is also possible to add potential matches to the candidate set that a grouping approach would not place together. For example, it is very unlikely that a bucketing approach would put the account "CoockingManic" in the same bucket as "Jan Janssen". However if there is any page on the Internet with the name "Jan Janssen" on it and also the account "CoockingManic" this makes it likely that Google search for "Jan Janssen" wil return this page and therefore the account is added to the candidate set.

5.3 Online entity resolution

5.3.1 Veldman

In 2009, Veldman performed research on matching internet profiles. See worked with data gathered from two major Internet sites: the Dutch social network site Hyves and LinkedIn. Entity resolution wasperformed on 1628 Hyves and 2158 LinkedIn profiles whith an high overlap. It was hypothized that performing entity resolution not only on the attributes of the entities, but also on their relationships with other users would improve the the results.

In a first phase, she established a baseline entity resolution performance on her dataset using a different text-based general-purpose comparison functions that only used attributes. This was called the *normal* approach. It was found that a persons name is the best discriminative attribute, followed by e-mailadress and date of birth. It is also mentioned that the latter are less often available. She also concluded that attributes regarding the education and/or (former) job descriptions were not very discriminative and could also lead to inpredicatable (fluctuating) results. She experimented with, but not limited to, the levenshtein distance (See also Section 5.4.1), the jaro distance and TF/IDF while matching strings. She concluded that Jaro distance based algorithms assigned the highest scores to matching names, however they were also likely to assign higher scores to non-similar names and thus lead to false positives. The levenshtein distance would assign almost as high scores to matching names and lower scores to

non-matching names, resulting in a little less recall but much better precision.

In a second phase, relationships to other entities were included while performing resolution. For each entity the links to other entities were gathered and the similarity of both sets was scored. This was the *network* approach. In this phase, the final score for every comparison was calculated based on weighting the *normal* score and the *network* score. These weighting was experimented with, however did not influence the outcome significantly. In the end, equal weights were assigned to both comparators. Against expectations it was concluded that the overall performance dropped slightly below that of the *normal* approach. Finally, in the third phase the network based approach was further extended by including the type of the relationship. However, this made the results drop even more.

Veldman speculated that this drop in performance was due to the fact that there were little ambigious profiles in the datasets. This speculation was supported by rerunning all experiments on a changed dataset where ambiguity was introduced delibratly. In the end it was concluded that including relationships to other entities can be used to improve precision, but is likely to decrease recall.

The difference between the work of Veldman and the IMatcher is that Veldman did explore relationships, whereas the IMatcher did not. Also, Veldman worked with prepared datasets, whereas the IMatcher works directly with the Internet.

5.3.2 Finding Nemo

In their technical report Jain and Kumaraguru [18] describe their attempt at developing an integrated system for linking different online social profiles of the same user. The novelty of their work is in the fact that they recognize these three different dimensions to do this: the social profile, content and connection network. By profile they mean personal details like name, date of birth, address, etc. Content consists of the attributes of user generated content like posts and upload. The connection network describes the number of friends, their names and possibly the type of connection. In their paper they provide an extensive discussion of related work and categorize it along these categories. In their own prototype a search engine / entity resolution model that finds Facebook accounts given Twitter accounts, they integrate three algorithms that each exploit one dimension.

Their approach differs from the one taken in this thesis by the fact that they start with one social profile, instead of the known details of a real world person. This provides them with the ability to extract all the attributes that they can from the input profile and using them for the search. For example, self-mentioning -where someone tweets a reference to one's own Facebook page- occurs quite frequently. This of course is not possible when starting with only real world details. Although, this might be exploited indirectly, where real world details would lead to the discovery of a Twitter profile, which in turn leads to the discovery of a Facebook profile. Another important difference between the work in

this thesis and that that of Jain and Kamuragur [18] is that they did not evaluate their approach on real Internet data but used a prefetched dataset, resulting in much faster searches.

5.3.3 Other approaches

Over the last two to three years, many other approaches towards using online profiles to gather data about individuals have surfaced. An example that recently got attention in the media is a system called Rapid Information Overlay Technologie [RIOT] [1]. This system can be used to quickly find people online, track there past activities and make predictions about future movements. It is important to note that there are still some manual actions that need to be performed, especially on the field of entity resolution, since RIOT lists a list of possible matches after a search. No metrics about the system are known.

Minder and Bernstein [24] report on using face-recognition to match profiles across multiple social networking sites. Although, there method in itself does not perform better than text based approaches, they do claim that combining both methods increases accuracy. They applied their approach to each combination of profiles in their pre-fetched datasets which consisted of 1610 and 1690 entries, with 166 matches. Which makes this a naive approach towards comparison function application.

A lot of attention has been paid to text-based comparison of profiles. For example Perito et al. [30] tried to use usernames from Google and EBay services, to find accounts that belong to the same user. They achieved an accuracy of 71%. An accuracy of 72% was achieved by Motoyama et al. [27], exploring attributes like full name, school, city, location and age to match identities. In their case name, school and e-mail address proved highly discriminating attributes. Minder and Bernstein [24] also included name, birth date and e-mail address in their text-based algorithm. They saw, when they added their face-recognition algorithm that the influence of e-mail address decreased and the influence of name and birth date increases.

Narayanan and Shmatikov [28] used a known, labeled network surrounding one user to de-anonymize a anonymous network surrounding the same user with an accuracy of 31%. Just as Veldman [36] concluded, a network based approach to identity resolution is likely to improve results. However, on it's own the results are often not as good as attribute based approaches. Another way a friends network or graph can be explored was researched by Labitzke et al. [21]. They worked under the assumption that people knew each other when there are at least 3 mutual friends.

Again, the main difference between this work and the IMatcher is that the IMatcher does not work with prepared datasets, but works directly with the Internet. This is not true for the RIOT system. Very little is known about this system, since there are no

[1] http://www.guardian.co.uk/world/2013/feb/10/software-tracks-social-media-defence

publications or reports about it, only a news item, however it seems that this system still requires some manual interaction whereas the IMatcher does not.

5.4 Applied techniques

This section provides a little more detail about some existing techniques that are used in the IMatcher. They are gathered here since they were incorporated without any change and are referenced from multiple places throughout this thesis.

5.4.1 Levenshtein distance

The Levenshein distance [23] is a metric to describe the similarity of two strings. The Levenshtein distance can be calculated recursive over two strings $A = \{a_1, ..., A_l\}$ and $B = \{b_1, ..., b_m\}$ as follows: $L(a, b) = max(L(a_1, ..., a_l - 1, B) + 1, L(A, b_1, ..., b_m - 1) + 1, L(a_1, ..., a_l - 1, b_1, ..., b_m - 1) + t)$ with $t = 0$ if$A_l = B_m$ and $t = 1$ if $A_l \neq B_m$. The recursion terminates if $|A| = 0$ or $|B| = 0$, resulting in $L = max(|A|, |B|)$. The result is the minimum number of edits needed to translate one of the strings into the other one. In this context an edit means either inserting, deleting or replacing one character.

The Java implementation is not recursive, but done in a nested loop, with a complexity of $|A| * |B|$. Finally, the number of edits needed to translate the strings is divided by the length of the longest string. This yields a normalized similarity score between 0 and 1 for strings A and B.

5.4.2 SteinhausJohnsonTrotter algorithm

The Steinhaus-Johnson-Trotter [19] is an efficient, non-recursive, in-place algorithm for determining all the permutations of a given set of integers. It is based on the observation that every permutation of length l, can be generated from every permutation with length $l - 1$ by inserting l at every position in every permutation. From this it follows that given a permutation, the next permutation can be calculated by swapping two elements. To calculate all permutations of the integers 1 to l, start with an order list of 1 to $l$and and for each integer set $direction \rightarrow left$. Find the largest integer i, where the adjacent number on it's direction is smaller then i and i is not the far left or right (equal to it's direction). If no such i exists, the algorithm terminates. If such an i exists, swap i with the adjacent number on it's direction and switch the direction of every integer greater than i.

In the IMatcher this algorithm is mainly used when multiple parts of an attribute are possibly ordered differently when comparing them. For example, the names "Jan de Boer" and "Jan Boer de" belong to the same human. However, a string comparison will

yield a low similarity score. To avoid this, a string comparison of the IMatcher consists of the highest of all scores for all combinations of permutations. To do this in Java, every string is exploded on whitespace into a number of parts, trimmed from further whitespace and comma's and than permutated. It can be seen that permutating the numbers 0 to $|parts| - 1$ and using the permutations as indexes when concatenating the parts again yields all combinations of partial strings. [1]

5.4.3 N-grams

N-grams form the basis of a technique that can be used to categorize texts, while not suffering from the drawback that it can be applied to only one (or a few) languages. It is even shown [10] that N-grams ca be used to identify the language of a text. N-grams provide very good results on texts of 50 characters or more and are shown to even perform on texts as short as ten characters [10]. The observant reader might notice that the IMatcher needs conciderably more characters than six. This is due to the fact that online language, and Twitter messages especially, often contain abbreviations or specific notation like hashtags to identify a topic(#computer) or intended users (@BravoBoy).

An N-gram is formed using n letters of a given given alphabet. For example, given the alphabet "a", "b", "c", "ab" and "bc" would be valid two-grams. Counting all n-grams in a texts provides the possiblity to predict the next letter in a series of n-1 letters or the next. Lets assume that the statistics in Table 5.1 were gathered from an example Dutch and English text. Giving a 3-gram that starts with "the red" there is a likely hood of over 50% that the next word will be "cross" if the language is English and 90% if the language is Dutch.

Table 5.1: Example 3-grams

The red ...

	English	Dutch
table	1	0
carpet	2	1
cross	3	9

The statistics in Table 5.1 can also be used to determine the change that a given 3-gram "the red carpet" is of either language. Assuming the language is Dutch, this model would predict the sentence correct *P("carpet" | "the red", Dutch)* = 1/10 = 10% of the time. Assuming the language is English, this model would predict the sentence correct *P("carpet" | "the red", English)* = 2/6 = 33% of the time. This makes it more likely

[1] The actual implementation is based on the suggestions provided at http://stackoverflow.com/questions/2920315/permutation-of-array

that the sentence "the red carpet" is English. Combining the results of predictions over all parts of the text, yields surprisingly good results predicting the language of a text. Please not that this is an example, for language detection n-grams of letters are used instead n-grams of words.

Chapter 6

Ethical considerations

This chapter discusses the ethical concerns surrounding the developed technology. Some ethical theory is discussed and both the experiments and application of the prototype are considered from an ethical point of view.

Already in 1998, Moor [25] argued for a new field of research: computer ethics. He argued that the current approaches to ethics were not applicable to the field of computer science. One of the reasons he put forward for this was the information enrichment that comes from using computers. The logical malleability of computers makes it able to constantly alter the function they perform. From here it follows that, among other things, information that is at some point processed by a computer, can from that point on be used in many ways. This means that automating tasks using computers, often results in a reduction of privacy. For this reason, Moor [25] was one of the first who argued that (intended) computer usage should be considered from an ethical point of view. Since then, more research has been performed with regards to the relationship between IT and ethics.

Currently, ethicists often take an existing phenomenon (which can be a policy, practice or technology) and judge in retrospect if it is ethically justifiable or not. One established school of thought is that of embedded values [12]. At the University of Twente, van Wynsberghe is currently researching the gap between ethics and the research and design of IT technologies. One direction of research is the development of a framework for incorporating ethics into the engineering cycle. The goal of van Wynsberghe her research is to develop a method to help guide the engineer towards a justifiable product while

developing it. As a first step in this research she proposed a number of guidelines for performing ethical justified research using social network sites.

Section 6.1 discusses ethical considerations surrounding data mining and social network sites research in general. Section 6.2 discusses the ethical aspects of the experiments conducted. Finally, Section 6.3 discusses whether the application of the prototype is justifiable or not. To do this, embedded values theory and the guidelines proposed by van Wynsberghe are discussed and then applied. Actually, the work presented in this thesis has played a significant role in developing these guidelines and is used as an example to illustrate their application.

6.1 In general

The developed prototype is mining data from social network sites. Although it can be argued that no data is really mined by the developed prototype, considering it in its intended context as part of a larger application that gathers attributes of people using their online manifestations does imply data mining. There are general ethical concerns considering data mining and surrounding social network sites which are discussed in the following two subsections.

6.1.1 Data mining

"Data mining offers automated discovery of previously unknown patterns as well as automated prediction of trends and behaviors" [20]. This implies that data mining technologies might yield unexpected, and therefore possibly unwanted, results. Dutch law allows gathering and processing personal information under a number of conditions, as stated in the Wet Bescherming Persoonsgegevens [1]. This law is the Dutch implementation of the European directive on data processing [2]. Three key points of this law are:

1. The subject of the data, should be fully informed about which information is gathered.
2. The subject of the data, should be fully informed about how the information is used and for what goal.
3. The subject of the data has the right to see all information that is collected about him, correct it or have it removed.

From the definition of data mining it follows that it is hard to fulfill this second condition. For example, stating on a customer loyalty card that all provided information will be used for marketing purposes is way to vague and not sufficient to comply with the law. If processing data to optimize floor plans, it should be stated that all purchases are

[1] http://wetten.overheid.nl/BWBR0011468

[2] http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31995L0046:en:HTML

analyzed to see if changes in floor plans influence consumer purchases. If transactions are used to compile user profiles and provide targeted weekly mailings, it should be stated as plainly as that. It is even more complicated to inform people upfront about unexpected results. For example, it is impossible to predict the results of mining association rules, so it is also impossible to fully inform the subject upfront.

It is also important to note that there is another law governing data gathering and processing by investigative authorities, the Wet Politiegegevens [1]. In short, this law suspends the right to be informed about the information gathering and processing when the subject is a suspect in an investigation. Despite that, information is still always gathered for a goal and cannot be used for anything else besides that goal. This is also in line with the European directive that provides governments with the right to gather data when it benefits its citizens. (That is, assuming that investigating felonies benefits society as a whole.)

6.1.2 Social Network Sites

Over the last few years, social network sites [SNSs] have become the subject of research. Social networks are a large source of information about persons. Henderson [14] concludes that researchers differ in opinion about usage of this data. For example, Lauren Solberg [34] concludes that "The Internet is a public space, and even with the password protections, security settings, and strict contractual terms of use that Facebook offers, Facebook users ultimately assume the risk that information posted on the Internet, and particularly on a social networking site, may become publicly available." Others claim that consent is needed from the subject before using data from SNSs. One argument for requiring consent is the relative ease with which data can be de-anonymized. For example, the data published in the Tastes, Ties and Time Facebook project has been reverse engineerd to identify the original subjects [2]. Finally, there are researchers that claim that it differs from case to case whether consent is needed [14].

The argument put forward by Solberg [34] is often voiced about information posted on the Internet, often called open data. However, it is not necessarily true. There is an established principle of expectancy of privacy [17]. Messages posted on a SNS are not as private as e-mail, just as talking to your friends in a bar is not as private as talking to them at home. When talking to your friends in a bar you should expect that people can overhear (parts of your) conversation, however should you expect people listening in on you on purpose? No, even when talking in a public space you can expect some level of privacy. The same line of reasoning holds online. When you post a message online, you should expect that people might come across it and read it. However, you do not have to expect that someone systematically gathers everything you ever posted and starts

[1] http://wetten.overheid.nl/BWBR0022463/

[2] http://www.michaelzimmer.org/2008/10/03/more-on-the- anonymity-of-the-facebook-dataset-its-harvard-college/

processing that data.

In response to this, one can argue that gathering information online is not done by hand, but automatically. For example, it is known that sites with houses for sale copy each others advertisements, only adding a link back to the original advertisement. Using this line of reasoning, it is stated, that no-one should expect any privacy online ever, since they are well aware of the fact that whatever they say can be gathered automatically. But this is a fallacy, not everything what *can* be done, *should* be done or is allowed. For example, it is technically feasible to put up telephone taps and record every telephone conversation of every Dutch citizen. However, this is not done (nor allowed) and there is a reasonable expectation of privacy on the phone. Therefore, it can also be excepted online to some (smaller) extent.

Another interesting view on using data from SNS, is that of copyright. There are not only laws governing copyright, which make it questionable to just copy and use data from SNSs, but there is also the moral question of whether it is appropriate to gather and duplicate conversations, pictures or any other type of expression from people without their permission. Even though copyright is often handed limited on SNSs, this does not necessarily mean that anyone can use everything posted on SNSs. Whether this type of reasoning holds, depends on the type of social network. For example, Facebook and Google+ build on the concept of sharing with your friends. Twitter, on the other hand, is about sharing with the world and has a policy of encouraging the re-use of tweets [1].

Since there are currently no established ethical frameworks for judging applications that use social network sites, Moreno et al. [26] propose a case-by-case analysis to determine if a specific research experiment is justifiable or not. This is done in Section 6.3.

6.2 About the experiments

This section discusses the experiments that were conducted and how it was made sure that they were performed ethically.

For the first experiment, full informed consent from all participants has been secured. subjects were included in the experiment only after self-selection and signing-up. There was no deception involved, so there was no risk for the subjects. Participants were informed that they would be researched by an automated system on, at least, Twitter and possibly other social networks. They were informed that data mining can lead to unexpected results, or none at all. All results are handed over to the subjects, so they can use the results to their own advantage. Finally, it was mentioned explicitly that subjects were allowed to leave during the experiment, so no unwanted discomfort should be possible at all. All in all this experiment was conducted fully in line with what can be expected from a moral researcher. See also Appendix B for more details concerning

[1] https://twitter.com/tos, chapter 8

subject selection.

For the second experiment, a dataset containing provided by the ISZW was used. These subjects are part of a group of temporary job agencies they are currently investigating as part of a risk analysis. Of the provided subjects, one third was classified as high risk, one third as low risk and one third as average risk. The researcher was not provided with the classification of each individual subject. The subjects were not (and will not be) informed about the research they are part of. For this experiment, and this experiment only, the research and his first supervisor were formally contracted by the ISZW as required by Dutch privacy law [1] to perform this experiment. So given this contract, the research performed was justifiable. Furthermore, investigative authorities often pilot a new type of research before going through the paperwork of getting it legislated in the laws that make privacy exceptions for investigative authorities.

For example, in 2005 the Inspectie Werk en Inkomen (now part of the ISZW) used water usage statistics from welfare receivers to identify people who received full welfare support, but did not live alone [31]. Since this new type of investigation was reported, as required by law, to the Dutch privacy agency, they started an investigation. In the end, the Dutch agency for enforcing privacy law did not approve of this approach and all fines issued because of this investigation were invalidated. However, Dutch judges overthrew this verdict and did legislate this type of investigation and many welfare subscriptions have been ended because lack of (or increased) water usage [2], entailing that the welfare receiver did not live at the declared address (or not alone), which violates welfare law. From this it can be concluded that new types of research, although reducing privacy are allowed on the conditions that they are reported and investigated by the privacy agency or legislated in new laws.

On a side note: At the faculty of of Electrical Engineering, Math and Computer Science [EEMCS] University of Twente there is an ethical committee in the process of forming. The committee is not formally installed yet and procedures are not final yet. For this research their preliminary guidelines have been followed and an application form was also filled. However, no approval or disapproval from the EC was received, so after a second inquiry subject selection and the experiments were started without. See also Appendix B for the original application to the Ethical Committee.

6.3 About the prototype

This section starts with a short discussion of value sensitive design in Section 6.3.1. This is followed by a short discussion of the guidelines proposed by van Wynsbherghe in Section 6.3.2. Finally these guidelines are applied to the developed prototype. This results in a judgment whether application of the developed prototype is justifiable as an

[1] Dutch: "bewerker" in de zin van de wet bescherming persoonsgegevens

[2] Dutch jurisprudence can be found at http://nl.vlex.com/tags/waterverbruik-981412

instrument for investigative authorities and in other contexts.

6.3.1 Value Sensitive Design

A relatively new, however well known approach to ethical design is Value Sensitive Design [VSD] [12]. The basis of VSD is formed by the observation that every piece of technology can have certain values embedded. In this context, a value can be anything "that is considered important by a group of people within a certain context" [12]. Embedded means that they are more likely to emerge from using the piece of technology than other values. For example, trackballs are designed as an alternative for computer mouses. Their design fits much better with the way the human body works and reduces the risks of rapid stress injury [RSI], therefore this design promotes the value of quality of life. On the other hand, guns are designed to shoot (f.e.) people, and therefore its design demotes the value of quality of life.

From this observation it follows that, an assessment can be made to discover which values a piece of technology promotes or demotes. Even more, while designing, embedded values can be discovered and their effects mitigated or strengthened. For example, the design of a chair could be extended with a timer that sounds when someone has been sitting for more then 30 minutes in the hope that they stand up and walk around for a bit. This would help reduce the chances of RSI and therefore promote the value of quality of life. VSD promotes to use conceptual, empirical and technical methods to identify stakeholders and which values might be emerge for them. With this, VSD encourages engineers to design good technology.

In a more extended view, van Wynsberghe [40] identifies that there are multiple ways values can become part of a design, namely assumptions and biases. For example, giving the packaging of dolls a feminine color is fueled by the assumption that the intended buyer of a doll is more likely to be a girl than a boy. Extensively using the colors red and green in a design might bias the design to non-colorblind people. (Which is a significant group of about 4% to 8% of the population [32]) It is important to note, that assumptions, by themselves, are not necessarily wrong. Still, assumptions and biases can provide important leads that result in discovering embedded values. This is a valuable tool, since predicting values at design time is more difficult than finding them through use after implementing the design.

An interesting observation is made by Akrick [2]. She says that engineers embed, besides values, another important decision into their designs. A design can imply the delegation of certain moral responsibilities from a person to a technology. For example, speed camera's eliminate any moral judgment about fining. Sometimes an officer can recognize moral reasons to not fine for a speeding, but only give a warning. A camera does not make such a judgment and always fines, so the responsibility to decide to fine or not, is delegated.

Finally, Verbeek [37] notes that it is likely that not all promised values will emerge. For

example, not all cars that look fast, will actually go fast. Or not every car that looks very robust, receives high scores in safety tests. For this reason, it is also important to do empirical research after launching a product to determine which values do actually emerge.

6.3.2 Guidelines for working with social network sites

As a first step in her research into incorporating ethics into engineering, van Wynsberge introduced a number of guidelines for working with data from SNSs [41]. The guidelines proposed by van Wynsbherge can be translated into a set of five questions that need to be answered by the researcher, preferably in collaboration with an ethicist. These five questions are:

- Who are the key actors? (Direct and indirect subjects, researchers etc.)
- What is the context and what does privacy mean in this context? (location and data content)
- What is the type and method of data collection? (active or passive)
- What is the intended use of information and amount of informationcollected?
- What are the intended values? (Value Analysis: making explicit and scrutinizing intended values of the researchers.)

Answering these five questions helps to develop a clear understanding of what the engineer wants to achieve, who will be affected by his work and how he wants them to be affected. It also helps to identify undesired effects that were not foreseen. This can help to guide the engineer to design safeguards that prevent possible misuse or minimize undesired effects through design. Of particular interest is the forelast question. It stimulates the engineer not only to describe what he wants to use the data for, but also what he does *not* want to use the data for. This helps to prevent gliding scale argumentation and finding new uses for data that "is already there anyway."

6.3.3 Applying the theory

What are the key actors? (Direct and indirect subjects, researchers etc.)

Besides the researcher himself, two other groups of actors were identified: those using the system and those that are affected by the system. The intended users of the system are the analysts of the ISZW and their supervisors. The persons who's Twitter profiles are crawled are the affected actors. This group should be divided into two subgroups. Figure 6.1 shows that searching for a set of online manifestations of a person, might yield both correct and incorrect online manifestations. This identifies two groups of subjects: the intended subjects and the innocent bystanders, in other words: the subjects who's online manifestations are searched and the people who's online manifestations are found but

are not the subject of the search. From now on these will be referenced as the intended subjects and the unintended subjects, together they are the subjects.

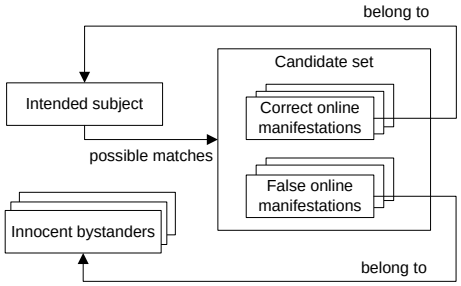


Figure 6.1: Candidate set, divided in correct and incorrect findings to identify intended and unintended subjects

What is the context and what does privacy mean in this context? (location and data content)

The context in which the prototype is working is Twitter. Twitter is a public social network that is about sharing with the world. This distinguishes it from other networks like Facebook, which is about sharing with your friends, or LinkedIn, which is about connecting with professional contacts. The goal of Twitter is to share your tweets with the world [1]. This does not necessarily entail that tweets can be used in any way a reader sees fit, however it also doesn't mean that Twitter users should expect that their tweets are protected.

What is the type and method of data collection? (active or passive)

For development of the prototype, all data of the intended subjects was collected with consent, this is what van Wynsberghe calls active. The intended subjects provided details about themselves to facilitate finding their Twitter. Also, they provided the name of their Twitter account as a ground truth for validation of the model. Application of the prototype in practice by the ISZW will be done without consent, even without explicitly informing the subjects. One rationale to allow this is that the ISZW has a legal duty to check if social support is only received by those who are entitled to it. Also, it is important to note that a method like this will not be deployed in practice without legislation.

[1] https://twitter.com/tos

What is the intended use of information and amount of information collected?

During the experiments the information is collected only to test and validate the model. After this, data will only be retained encrypted for a period of time due to legislation surrounding master thesis grading. The information is explicitly not collected for commercial use or to learn personal details about the subjects lives. If the developed prototype would be deployed in practice, the data would be collected only to determine if a subject has a Twitter account and if yes, which one. Of course, after identification of a Twitter account, the found account would be monitored by the ISZW to search for signals of fraud like living together unreported, undeclared income or undeclared savings.

The exact data collected from each tweet are the used language, e-mail addresses and telephone numbers. If the Tweets had a geotag, these were retrieved to. Further extractions that have been considered, but moved to future work, include: determining location from the tweet text, guessing the subject of the tweet, gathering retweets, gathering followers and those followed. From each profile the name, location, profile picture, URL and free description are extracted.

What are the intended values? (Value Analysis: making explicit and scrutinizing intended values of the researchers.)

For the direct users of the developed prototype the expected result is that they have more characteristics of the persons they are currently performing risk analysis on. This should result in better results and therefore increase self-fulfillment from their jobs. By extension the officials that carry out risk analysis and physical inspections, work for society. So, although no real users, they are -through elections and politicians- employed by society to battle fraud. With the new prototype, it is expected that officials can work more effective. Gathering more characteristics of subjects, will improve the results of risk analysis and therefore increase the chance of discovering fraud. Discovering fraud will result in fewer people receiving social support and thus less spendings on social security. Less people receiving social support, while not being entitled to it promotes the values of fairness and justice in the whole of society. Also, spending as little as possible is for almost anyone a value as well.

For all subjects the value of privacy is demoted: Information about them is gathered and stored in a systematic way. This information is gathered with the intent to process it and to derive new information from it. This information makes it possible to see where someone has been and if enough updates are available: to follow someone around. Even more, this information can be used to make predications about where someone will be in the future. All these are different aspects of privacy, the right of every individual to decide which personal information to make available to anyone else [29].

Working with online manifestations abstracts away from the person behind these manifestations. This strongly embeds the disvalue of objectification into the system. Viewing

persons mere as a set of online profiles and the characteristics that can be derived from them might trigger an unwanted attitude towards the data. Analysts might easier share data about "someone online" than "Jan Jansen from Enschede"

Finally, for the intended subjects two other possible consequences were discovered: accountability and calmness. Although not really values in the sense of value sensitive design, they are important consequences. Calmness is demoted in fraudulent subjects. Since there is yet another way their fraud can be discovered and they might get more and more nervous about being discovered. On the other side, for non-fraudulent users accountability is increased. They know they are rightfully receiving support and they have yet another type of investigation that proofs so. They might feel even more accountable and less defensive about their social support towards other people.

Assumptions and biases
As stated in the section before, value sensitive design urges engineers also to think about assumptions and biases. This section explicitly mentions the assumptions and biases that were identified.

Two of the most important values that are expected to emerge, fairness and justice, are based on the assumption that everyone in society will actually value this positive. It is difficult to argue on behalf of a whole country, however there are two reasons that this seams reasonable. First of all, part of VSD is the assumption that some values are universal. For example, respect for human life is a value that is easy to accept as universal and shared between many cultures. Furthermore, fairness and justice are the values that form the basis of social law. Not valuing them in execution of these laws, would contradict itself. Therefore, this assumption seems justifiable.

The proposed prototype and its application does carry a bias towards people that have an active social life online. These people expose more of themselves offline and are therefore more likely to be found online. Since being found online is the first step in receiving a predication about being fraudulent or not, this group is more likely to receive a prediction. Although these subjects are more likely to receive a prediction, the prediction itself is not influenced by the fact that they are more active online.

Extending this line of reasoning, provides another bias that influences the prediction. Some types of fraud depend on gathering attention towards yourself. For example, undeclared income through repairing and selling second hand bikes requires getting customers. Another example is that of temporary job agencies that provide cheap painters; they need to get the news about their service "out there" for people to hire painters through them. People who are either (i) not informed about this type of fraud detection or (ii) are not capable of grasping its implications are more likely to be detected than those performing more covert types of fraud.

6.3.4 Justifiable or not?

Now all is in the open, which was the goal of answering these questions, it has to be determined if development of a prototype for entity resolution as part of fraud detection is justifiable or not. The main incentive of the researchers and also the ISZW is to protect the financial and social security of the Dutch state. Although fairly abstract, this is an important goal. Not only for those paying taxes that enable social security, also for those receiving support. Making sure that only those who really need it receive support, creates more support under tax payers for social law than just "handing out money." Also, it is important to consider that the information on Twitter is not supposed to be considered as private as a telephone conversation or private message to a friend. For these reasons, the value trade-off between security of the state and privacy is justified. In other words, the use of data from online SNSs for this experiment and application falls within ethical limits.

Stating it like this also implies that if that trade-off is not in place using a prototype like this is not ethically justifiable this way. In other words, if the goal is not to protect the security of the state but, for example, commercial, using this type of technology is not necessary justifiable.

6.4 Conclusions

In this chapter a number of ethical aspects of the conducted research were discussed. First of all, the ethics of research around data mining and social network sites were discussed. Both lead to the conclusion that any research using any of them needs to be assessed on a case-by-case basis to see if it is the right thing to do. After this, the research itself was considered. Both experiments were considered in detail and judged sufficiently justified to be carried out. Finally, the developed prototype and its intended application were analyzed. Using a set of guidelines proposed by van Wynsbherge it was concluded that applications in a context that really benefits the whole of society is justifiable, whereas other applications might not be.

Finally, doing an analysis like this is not a common thing in computer science research. However, I do believe that it is an important thing to do. Computer science is a relatively new field of research and, compared to some other sciences, still immature. For example, comparing computer science to medical science yields an important parallel and also shows that both fields deal with them differently. Both medical and computer science share that discoveries in their field have a major impact on the world as we know it. The changes in society that have occurred over the last decades due to both medical and informatics research are so massive that it is almost impossible to conceive a world without them. However, both fields differ strongly in the approach they take towards considering the consequences of these discoveries. I believe that we, the IT community, can learn from the medical field, which also had to learn to consider ethical aspects of

their discoveries.

I realize, medical decisions often have to do with life and death and the quality of life. However, changes induced by IT have also had drastic consequences. More and more cases of cyberbulling, sometimes resulting in sewerside are reaching the media. Other, less dramatic effects include social isolation or identity theft. These kind of developments force us to address the ethics related to our work as computer scientists, as engineers.

Chapter 7

Conclusion

This thesis answers the question if it is feasible to find someone's Twitter account given some personal information about him or her. This was explored partially at the request of the Dutch social investigative authority [ISZW]. They are currently exploring the possibilities for adding information from online sources to their risk analysis.

The approach chosen in this thesis, is based on a three step process. First a set of heuristic queries using known information is executed to find possibly matching accounts. Secondly, all these accounts are crawled and information about the account, and thus its owner, is extracted. Thirdly, all possible matches are examined and the correct account is determined.

Before turning back to the main question posed, the subquestions posed in Section 1.3 are revisited:

> *Which requirements would investigative authorities, like the ISZW, pose on an application for online entity resolution? And can these requirements be fulfilled?*

The main goal for the ISZW is finding more characteristics about subjects. The two most requirements that follow from this are the absence of any bias and correctness of the results. For investigative authorities in general it is important that their investigations have no bias against gender or ethnicity and to make their risk analysis worthwhile information needs to be of high quality and thus correct.

Due to the low number of subjects it is hard to make any statement regarding bias towards certain groups. From the subject no information about ethnicity, religion or race was known at all. However, there is no reason to suspect that the chosen approach is sensitive to any of these criteria. Regarding correctness of the results; a regression analysis of the results of experiment 1 showed that based on the IMatcher results showed no false reports. In other words, for every subject where the classifier claimed that the candidate account was correct it actually was correct. Therefore, correctness of the results, for this set of subjects, was 100%.

For more details on the requirements, see Section 2.1. For more details on the experiments, see Section 3.1.

Which data is available on online social networks and other online manifestations?

The information available differs from social network to social network. The developed prototype was limited to Twitter which offers a number of fields that the user can fill out: name, location, URL, description. All these fields are free format, so users are not obliged to provide information in a standardized form or at all. Furthermore, every account has a number of tweets with more information attached. Information contained from a tweet can be the language, the location, the time of the tweet, e-mail address, telephone number, the subject, hints about age or gender. From all this information, the IMatcher currently only extracts geo tagged locations, language, e-mail addresses and telephone numbers.

Which data about the real world person is minimally needed to reliably match an online manifestation with a real world person?

To answer this question, four variations of experiment 1 were ran. The first variation included all know details about the subjects, the second variation did leave out the last names, the third variation left out all addresses and the fourth variation left out all e-mail addresses and telephone numbers.

Comparing the results the the variation with full information and the variation without last name it can be concluded that the last name are required. Leaving out only the last name the percentage of correct Twitters that is included in the candidate sets drops from 58% to 17% when exploring the top 8 of all heuristic queries and from 92% to 25% when exploring the top 20 of all heuristic queries. Therefore, having last name is crucial for completeness. This conclusion is also supported by the fact that the one account not found, did not include the last name, making this likely the reason it was not found. Leaving the last name out has a mixed influence on finding the correct Twitter account in the candidate set. For the first 3 runs the results looked slightly better, however they dropped down after these 3 runs.

Comparing the results of the variation with full information and the variation without addresses showed that leaving the address of the subject out of the search resulted in

exactly as many correctly identified Twitter accounts included as with the address used. It can also be seen that the candidate sets are considerably smaller, about 2 to 3 times as small. Therefore, removing addresses from the search does not reduce inclusion, but does reduce the number of wrong results. However, leaving the location out of the classification results in worse results than with location information.

Finally, comparing the variation with full information and the variation without e-mail addresses and telephone numbers shows that leaving out e-mail addresses and telephone numbers has no influence on the results at all.

In the end, the different variations showed that leaving only the last name is an crucial field to include in the search. However, addresses do help to better distinguish the correct Twitter account in search results.

More details on the experiment 1 and it's variations are provided in Section 3.1.

> *What is the relation between the time allowed for resolution of entities and the accuracy of the results?*

The results for experiment 1 show that no improvement at all over time, not for any of the variations. This can be explained by the fact that name is currently the best discriminating attribute, which is extracted from the Twitter profile itself, not from the Tweets. Since profile information is not changed very often it is not surprising that a change in profile information gets an account to be included in the heuristic search. Even more surprisingly, the regression results worsen over time. This is most likely due to the fact that the number of incorrect Twitter accounts increases every run. The number of correct accounts however stays the same, which makes it on average more an more likely for any account to be wrong. This results in classification algorithms developing a preference for classifying every account as wrong.

This seems to indicate that the expectation that accounts might sometimes pop up in query results to be included in the candidate set does not hold. However, this might be due to the small scale of the experiments. Longer running experiments on more subjects might encounter this phenomena and benefit from it.

> *Which characteristics of a real world person can be extracted, when these online manifestations are found?*

Except for the basic attributes mentioned earlier, like username, full name, age, gender, there are also more high-level attributes that can be extracted. For example, attitude towards society or politicians might be extracted if enough tweets or online posts are available. These might turn out to be strong predictors about the likelihood that the subject is fraudulent.

Given a limited, and possibly incorrect or incomplete, set of data about

a real world person, can their online manifestations be found reliably and automatically?

Based on these experiments, the answer is yes. The first, heuristic, search to gather a number of possible matches, works quite well. Of the 22 subjects, 12 had a Twitter account, of which 11 were retrieved using the heuristic search, which is 92%. Widening the heuristic search even more might also have returned the one accounts that was missed in the experiments that were run. So the correct account is very likely to be discovered, although amongst a lot of wrong accounts: noise. And there's the downside, the current prototype is not capable of distinguishing correct from incorrect accounts. yet. Nor can it reliably determine yet a subject even has a Twitter account at all. However, there is much ongoing research into feature extraction from tweets, so it is only a question of time before this hurdle is overcome as well.

All-in-all, this set of experiments have show that it is possible to perform entity matching on people and Twitter accounts. There are still many challenges to overcome, however this is a viable approach in practice.

Making the circle complete and returning to the ISZW, finding characteristics about subjects online and including them into their risk analysis seems viable. Before starting this project, one of the most important challenges for them to overcome was online identity matching, for which it is now show that there is no reason to believe that this is not possible.

Research on subjects like this also has an ethical side. Whether we can do this is important, but we should also ask ourselves if we should want to. This question was explored in Chapter 6 where it was concluded that such an approach is justifiable for fraud detection by governments, but not necessarily for other applications.

7.1 Recommendations

To make a model like the IMatcher a success, a number of changes have to be made. More-or-less in decreasing order of importance, the following improvements are proposed:

- First and foremost, the scalability issues that have been discussed in Section 4.3 have to be addressed. Most issues mentioned can quickly be addressed to make the IMatcher run faster and be more scalable. The only reason this was not done yet is to make all measurements comparable.
- Although the correct Twitter account is currently very likely to be included in the candidate set, the current approach to identifying this correct account using machine learning techniques has proven insufficient. For this, more characteristics of Twitter accounts have to be extracted and compared to what is know about the real world person. There are many techniques available for extracting specific characteristics. Incorporating these might greatly benefit finding correct accounts

in candidate sets. There is much ongoing research regarding this question, so there is no reason to believe that this cannot be done

- Extending the IMatcher to include other social networks can have a number of beneficial effects. If for some people their Twitter cannot be found, it might be that their Facebook, LinkedIn or Marktplaats account can be found. If even just one of these four accounts can be found, some characteristics might be extracted that can be used in risk analysis. Also, other research has shown that there is a high chance that people self-reference their profile on other networks, so finding one online manifestation might lead to finding more online manifestations.

References

[1] S. Abiteboul, B. Kimelfeld, Y. Sagiv, and P. Senellart, "On the expressiveness of probabilistic xml models," *The VLDB Journal*, vol. 18, no. 5, pp. 1041–1064, 2009.

[2] M. Akrich, "The de-scription of technical objects," *Shaping technology/building society*, pp. 205–224, 1992.

[3] R. Ananthakrishna, S. Chaudhuri, and V. Ganti, "Eliminating fuzzy duplicates in data warehouses," in *Proceedings of the 28th international conference on Very Large Data Bases*. VLDB Endowment, 2002, pp. 586–597.

[4] H. Been, "Catching welfare fraudsters online," *Research Topics report, University of Twente*, 2012.

[5] ——, "Entity resolution using search broadening," *Paper for the XML and DB 2 course, University of Twente*, 2013.

[6] O. Benjelloun, H. Garcia-Molina, D. Menestrina, Q. Su, S. Whang, and J. Widom, "Swoosh: a generic approach to entity resolution," *The VLDB Journal*, vol. 18, no. 1, pp. 255–276, 2009.

[7] P. Bosc and O. Pivert, "Modeling and querying uncertain relational databases: a survey of approaches based on the possible worlds semantics," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 18, no. 05, pp. 565–603, 2010.

[8] D. Brizan and A. Tansel, "A survey of entity resolution and record linkage methodologies," *Communications of the IIMA*, vol. 6, no. 3, pp. 41–50, 2006.

[9] CBS, "Aantal bijstandsuitkeringen blijft groeien," *CBS Persbericht PB13-015*, 2013.

[10] T. Dunning, *Statistical identification of language*. Computing Research Laboratory, New Mexico State University, 1994.

[11] I. W. en Inkomen, "Signaleren van fraude," 2009.

[12] B. Friedman, P. Kahn, and A. Borning, "Value sensitive design: Theory and methods," *University of Washington technical report*, pp. 02–12, 2002.

[13] P. Hartog, A. Molenkamp, and J. Otten, *Kwaliteit van administratieve dienstverlening: managen is integreren.* Kluwer, 1992, no. 1.

[14] T. Henderson, L. Hutton, and S. McNeilly, "Ethics in online social network research," *http://torrii.responsible-innovation.org.uk/case-studies/ethics-online-social-network-research-0; accessed at 28 March 2013*, 20129.

[15] M. Hernández and S. Stolfo, "Real-world data is dirty: Data cleansing and the merge/purge problem," *Data mining and knowledge discovery*, vol. 2, no. 1, pp. 9–37, 1998.

[16] C. Hofmann, E. Horn, W. Keller, K. Renzel, and M. Schmidt, "The field of software architecture," 1996.

[17] L. D. Introna, "Privacy and the computer: why we need privacy in the information society," *Metaphilosophy*, vol. 28, no. 3, pp. 259–275, 1997.

[18] P. Jain and P. Kumaraguru, "Finding nemo: Searching and resolving identities of users across online social networks," *arXiv preprint arXiv:1212.6147*, 2012.

[19] S. Johnson, "Generation of permutations by adjacent transposition," *Mathematics of computation*, vol. 17, no. 83, pp. 282–285, 1963.

[20] C. Kleissner, "Data mining for the enterprise," in *System Sciences, 1998., Proceedings of the Thirty-First Hawaii International Conference on*, vol. 7. IEEE, 1998, pp. 295–304.

[21] S. Labitzke, I. Taranu, and H. Hartenstein, "What your friends tell others about you: Low cost linkability of social network profiles," in *The 5th SNAKDD Workshop 2011 on Social Network Mining and Analysis*, 2011.

[22] S. Lee, "Imprecise and uncertain information in databases: An evidential approach," in *Data Engineering, 1992. Proceedings. Eighth International Conference on.* IEEE, 1992, pp. 614–621.

[23] V. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," in *Soviet physics doklady*, vol. 10, 1966, p. 707.

[24] P. Minder and A. Bernstein, "Social network aggregation using face-recognition," 2011.

[25] J. H. Moor, "Reason, relativity, and responsibility in computer ethics," *Computers and Society*, vol. 28, pp. 14–21, 1998.

[26] M. Moreno, N. Fost, and D. Christakis, "Research ethics in the myspace era," *Pediatrics*, vol. 121, no. 1, pp. 157–161, 2008.

[27] M. Motoyama and G. Varghese, "I seek you: searching and matching individuals in social networks," in *Proceedings of the eleventh international workshop on Web information and data management.* ACM, 2009, pp. 67–75.

[28] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in *Security and Privacy, 2009 30th IEEE Symposium on.* IEEE, 2009, pp. 173–187.

[29] Y. Onn, M. Geva *et al.*, "Privacy in the digital environment," *Haifa Center of Law & Technology*, pp. 1–12, 2005.

[30] D. Perito, C. Castelluccia, M. A. Kaafar, and P. Manils, "How unique and traceable are usernames?" in *Privacy Enhancing Technologies.* Springer, 2011, pp. 1–17.

[31] C. B. Persoongegevens, "Bevindingen ambtshalve onderzoek waterproof," Den Haag, The Netherlands, 2007.

[32] R. Pickford, "Natural selection and colour blindness," *The Eugenics Review*, vol. 55, no. 2, p. 97, 1963.

[33] N. Reelick, "Risicoprofielen en het opsporen van fraude bij een wwb-uitkering," *Journal of Social Intervention: Theory and Practice*, vol. 19, no. 1, pp. 60–76, 2010.

[34] L. B. Solberg, "Data mining on facebook: A free space for researchers or an irb nightmare," *U. Ill. JL Tech. & Pol'y*, p. 311, 2010.

[35] M. Van Keulen, A. De Keijzer, and W. Alink, "A probabilistic xml approach to data integration," in *Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on.* IEEE, 2005, pp. 459–470.

[36] I. Veldman, "Matching profiles from social network sites," *Masther's thesis, University of Twente*, 2009.

[37] P.-P. Verbeek, *What things do: Philosophical reflections on technology, agency, and design.* Penn State University Press, 2005.

[38] D. Verhue, B. Koenen, and R. van Kalmhout, "Issuemonitor q2 2012 fraudebestrijding," Amsterdam, The Netherlands, 2004.

[39] S. Whang, D. Menestrina, G. Koutrika, M. Theobald, and H. Garcia-Molina, "Entity resolution with iterative blocking," 2008.

[40] A. Wynsberghe, "Designing robots with care: creating an ethical framework for the future design and implementation of care robots," 2012.

[41] A. v. Wynsberghe, H. Been, and M. v. Keulen, "To use or not to use: guidelines for researchers using data from online social networking sites," *Responsible Innovation in ICT*, Accepted for publication on 20 May 2013.

Appendix A

Sink details

This appendix provides a list of all sinks developed for the IMatcher. They are grouped by function into sinks that find candidate matches, sinks that crawling candidate matches and finally sinks that assist in performing entity resolution.

A.1 Discovering candidate matches

Name:	**KnownTwitterIterator**
Input-type:	Person
Manipulation:	Adds candidate matches
Output-type	Person
Side-output-type:	TwitterAccount

This sink is solely responsible for iterating all twitter accounts that were found in an earlier crawl session. Iterating these accounts is necessary to make sure that all possible manifestations are entered into the pipeline that processes all Twitter accounts for feature extraction. This sink is also responsible for pruning unlikely candidates. This is done in three steps. Accounts with an average similarity score below 0.1 are pruned after 5 crawls, accounts with an average similarity below 0.2 are pruned after 10 crawls and accounts with an average similarity score below 0.3 are pruned after 15 crawls. These thresholds did not resulted in few accounts being pruned. This was done deliberately to

make sure that no correct accounts were pruned, which would pollute the classification analysis.

Name:	**TwitterByGoogleFinder**
Input-type:	Person
Manipulation:	Adds candidate matches
Output-type	Person
Side-output-type:	TwitterAccount

This sink is responsible for finding candidate matches. It takes the name, known aliases, e-mail address and telephone number from the person and executes a number of Google queries like "Twitter *firstname lastname*", "Twitter *firstname*", "Twitter *aAlias*", etc is build. A Neogeo workflow is used to execute each query, paginate the results and fetch the results one by one. Currently, the four top 20 results are processed. This was changed at 3 May. Before 3 May, only the top 8 results are processed. At this moment the amount of queries was also doubled. For each query "Twitter xyz" another query "xyz site:twitter.com" was added.

Each result of the queries is examined and if it is a valid Twitter account, added to the database as a candidate match for the input person. If the result itself is not a Twitter account, the result page is retrieved and the full HTML of that page is examined to see if it contains one or more Twitter accounts. If it does, they are added as possible accounts as well. This filter makes use of the static TwitterUtils class to determine if a URL is a valid Twitter account and extract Twitter accounts from HTML.

Name:	**TwitterByLocationFinder**
Input-type:	Person
Manipulation:	Adds candidate matches
Output-type	Person
Side-output-type:	TwitterAccount

This sink is responsible for finding candidate matches. If for the person one or more locations are known it performs a search for all tweets within 200 meters of each address. For all tweets found, the sender is retrieved and added to the candidate set. The disadvantage of this filter is that it can produce high numbers of possible candidates. On average this searcher finds around twenty new accounts the first day it is run and considerably less each following day. However, for some locations, there is a much higher number of candidates returned. For example, addresses which are in the center of a city also attracts many visitors, and in this case many candidates are generated. To counter this effect, unlikely accounts are pruned by the KnownTwitterIterator.

Name:	**PersonPersister**
Input-type:	Person

Manipulation:	Updates this persons representation in the database
Output-type	Person
Side-output-type:	-

This sink is meant to be the last sink in a Person pipeline. This sinks updates the current representation of this person in the database with new information. This are mainly new candidate matches for online presences.

A.2 Crawling candidate matches

Name:	**TwitterGeneralInfoExtractor**
Input-type:	TwitterAccount
Manipulation:	Extracts the profile name, url and description from the account
Output-type	TwitterAccount
Side-output-type:	-

Name:	**TwitterPictureExtractor**
Input-type:	TwitterAccount
Manipulation:	Extracts the profile picture from the account
Output-type	TwitterAccount
Side-output-type:	-

Name:	**TwitterMessgesExtractor**
Input-type:	TwitterAccount
Manipulation:	Enters all tweets into the Tweet pipeline
Output-type	TwitterAccount
Side-output-type:	TwitterMessage

Name:	**GeneralMessageLanuageExtractor**
Input-type:	Any Uncertain Object (f.e. a TwitterMessage)
Manipulation:	Determines the language of the given text
Output-type	Any Uncertain Object (f.e. a TwitterMessage)
Side-output-type:	-

Name:	**GeneralMessageEmailTelephoneExtractor**
Input-type:	Any Uncertain Object (f.e. a TwitterMessage)
Manipulation:	Extracts all e-mail addresses and telephone numbers
Output-type	Any Uncertain Object (f.e. a TwitterMessage)
Side-output-type:	-

Name:	**TwitterGeoExtracter**
Input-type:	TwitterMessage
Manipulation:	Extras geo information from a tweet
Output-type	TwitterMessage
Side-output-type:	-

Name:	**TwitterMessagePersister**
Input-type:	TwitterMessage
Manipulation:	Persists a TwitterMessage
Output-type	TwitterMessage
Side-output-type:	-

This sink is the last sink in the TwitterMessage pipeline. It saves every message to the database.

Name:	**TwitterAccountPersister**
Input-type:	TwitterAccount
Manipulation:	Updates this Twitter accounts representation in the database
Output-type	TwitterAccount
Side-output-type:	-

This sink is meant to be the last sink in a TwitterAccount pipeline. It saves the updates representation of the TwitterAccount passing through back to the database. Updates are can include new possible values for attributes or more support for certain values.

A.3 Entity resolution

Name:	**TwitterPossibilitiesAdder**
Input-type:	TwitterCompareAssignment
Manipulation:	Adds all possible matches to the compare assignment
Output-type	TwitterCompareAssignment
Side-output-type:	-

This sink is the first sink in the matcher pipeline. It enumerates all possible matches and adds them to the compare assignment, so they can be compared to the known details about the person.

Name:	**TwitterNameContainmentMatcher**
Input-type:	TwitterCompareAssignment
Manipulation:	Compares the name of the person and account
Output-type	TwitterCompareAssignment
Side-output-type:	-

This sink calculates the score nameContainment. It does this by determining how much of the known name appears in the name of the twitter account. This is done by enumerating all first names, tussenvoegsels and lastnames of the known person and determining if they are (in any way) part of the name provided on the Twitter account using String.indexOf(). The result is divided by the total count of first names, tussenvoegsels and lastnames.

Name:	**TwitterNameLevenshteinMatcher**
Input-type:	TwitterCompareAssignment
Manipulation:	Compares the name of the person and account
Output-type	TwitterCompareAssignment
Side-output-type:	-

This sink calculates the score nameLevenshtein. It does this by enumerating all permutations of first names, tussenvoegsels and last name. Each permutation is than compared to the name given to the account using the Levenhstein distance. This edit distance is then translated to a similarity score and the highest score is returned as the nameContainment.

Name:	**TwitterLanuageMatcher**
Input-type:	TwitterCompareAssignment
Manipulation:	Compares the language of the person and account
Output-type	TwitterCompareAssignment
Side-output-type:	-

Name:	**TwitterLocationMatcher**
Input-type:	TwitterCompareAssignment
Manipulation:	Compares the location of the Twitter account to know physical addresses
Output-type	TwitterCompareAssignment
Side-output-type:	-

This sink calculates the scores country, city, street and houseNumber as listed in Table 2.2, describing the similarity of the person and each possible match on these four levels.

Name:	**TwitterGPSMatcher**
Input-type:	TwitterCompareAssignment
Manipulation:	Compares the location of the Twitter account to know physical addresses
Output-type	TwitterCompareAssignment
Side-output-type:	-

This sink calculates the scores distance and distanceClose as listed in Table 2.2, describing the similarity of the person and each possible match on these two aspects.

Name:	**ScoresCummulator**
Input-type:	TwitterCompareAssignment
Manipulation:	Calculate the average of all scores
Output-type	TwitterCompareAssignment
Side-output-type:	-

Calculates the average of all earlier scores. Scorers that provided null instead of a score are ignored.

Name:	**TwitterMatcherResultsPrinter**
Input-type:	TwitterCompareAssignment
Manipulation:	Prints all results
Output-type	TwitterCompareAssignment
Side-output-type:	-

This sink is the final sink in the matcher pipeline. It prints the results in a CSV format that is suitable for use in a classifier.

Appendix B

Prosal for the Ethical Committee

The following pages provide the orignal proposal that was sent to the Ethical Committee of the faculty to gain permission for the experiment

Identity resolution: Finding online manifestations of a real world person

Henry Been
Supervisors: Maurice van Keulen, Pascal van Eck
20 November 2012

Introduction

In The Netherlands, over the last years, more and more pressure is put on investigative authorities to pursue a signal-driven approach to fraud detection. This means that database and domain specialists try to determine which group are most likely to commit fraud so inspectors can focus on these groups. Such a signal-driven approach is facilitated by compiling high risk profiles and finding persons that match these profiles. Until now this approach is not as successful in practice as is deemed necessary. The Dutch social investigative authority claims that this is due to not having enough features that can be used to classify persons as either fraudulent or not (Reelick, 2010). Furthermore, they speculate that on the Internet in general, and online social networks in particular, more information is available to extract more features (Inspectie SZW, 2010). Therefore they are interested in investigating if more features can be gathered online to classify persons as possible fraudsters or not.

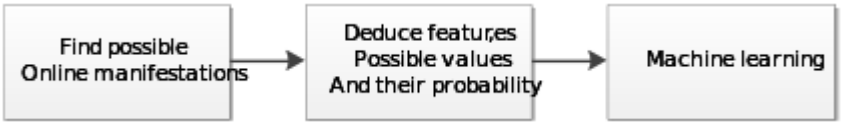


Figure 1: Signal-generation from Internet data

If possible, such a process would be performed in three steps, see also Figure 1: (i) Finding possible online occurrences of a person, (ii) Deducing features and possible values and their probability from the possible occurrences and (iii) Performing machine learning on the resulting features. The key challenge in steps two and three will be information retrieval [IR] and dealing with uncertainty in the input and incorporating this in existing algorithms for IR and machine learning. This are partially solved problems with ongoing research in the respective fields. It is believed that the performance of the current state of the art on this fields is good enough for application and that future developments can only enhance the performance during these two steps. The first step is assumed to be the “make or
break" for the complete system. This needs to be done “good enough” and “quickly enough” to have a practical value in investigations. What exactly is good enough will be explored together with investigative authorities, but the biggest foreseen risk is finding to much more-or-less matching manifestations, introducing to much noise into the rest of the process.

To investigate this, the main question this research will answer is:

Given a limited, and possibly wrong, set of data about a real world person, to what extend can their online presences be found reliably and automatically?

Answering this question will help answering the question currently faced by investigative authorities: is the Internet a viable source of information for signal-driven fraud detection? If the answer is yes, it is expected that further research will be conducted to develop a full system that performs all three steps.

Research method

The research question will be answered by developing a prototype for finding online manifestations of a given set of individuals. Experiments will then be conducted, by changing the input, to see how well the prototype performs under certain conditions. The input will be changed to investigate the influence of missing or wrong attributes on the results.

Prototype

A prototype application will be developed for performing identity resolution. The input for this application will be some personal details about the subjects. Example details are first name, last name, place of living, country, language, e-mail addresses, but more can (and will) be included. The application will use these details to search for online manifestations of subjects using proprietary API’s like the ones provided by Twitter, Google, Facebook, etc. Any possible manifestation will be recorded and from that point on be crawled on a daily basis. Crawling these possible manifestations will, hopefully, every now and then provide an hint that can be used to determine if the possible manifestation is actually a real manifestation or not. For example, when a known e-mail address or phone number is found on a possibly matching Twitter account it is more likely that this account belongs to the searched subject.

Experiment

Baseline performance will be established by running the application for a certain amount of time. More experiments will run on changed data to see how the application will perform when the input data is not completely correct. For example, will providing a wrong first name influence the number of presences found or the believe that certain presences are correct . For both baseline performance as experimental performance the results will be calculated at different intervals, for example after 1 day, 1 week, 1 month, etc. This will allow for establishing if there is a connection between how long a subject is searched for and the amount and correctness of presences found.
The two experiments will be run in parallel to make sure they are both based on the same state of the world since ranking on keywords by different API's cannot be assumed to be constant over a period of weeks or even months.

Subject selection
To experiment with the prototype, it is necessary to work with real existing people. Only working with real subjects, selected in an a-select manner, will provide results that can be generalized to a more general population. To facilitate this, subjects will be recruited on a website where they will be fullyinformed about the goal and mechanics of the research. The added value for the participants will be a full report on what the prototype has found about them, allowing to assess their own online privacy.

References

Inspectie SZW (2010). Zwarte fraude beter bestrijden, s.l.: s.n.
Reelick, N. (2010). Risicoprofielen en het opsporen van fraude bij een wwb-uitkering,
Journal of Social Intervention: Theory and Practice, 19(1), pp. 60–76.

3. Checklist for the principal researcher when submitting a request to the EC or the EC member for an assessment of the ethical permissibility of the proposed research

3.1 General

When answering the questions, it is advisable to consult the chapter on standardized research because the answers will be considered with this in mind.

1. Title of the project: *Identity resolution: Finding online manifestations of a real world person*
2. Principal researcher (with doctoral research also a professor): dr. *Maurice van Keulen, dr. Pascal van Eck.*
3. Researchers/research assistants (doctoral candidates, students etc. where known): *Henry Been, BSc*
4. Department responsible for the research: *EWI/DB*
5. Location where research will be conducted: *There is no physical location where the experiment is conducted.*
6. Short description of the project (about 100 words*): In a controlled experiment a prototype for finding online manifestations of real world persons is evaluated. It is investigated to what extend correct (and only) correct manifestations can be found given limited an possibly partially wrong information about a real world person.*
7. Expected duration of the project and research period: Starting *November 2012 up to May 2013*
8. Number of experimental subjects: *as many as sign up, ideally at least 50*
9. EC member of the department (if available): *Dr Pascal van Eck*

3.2 Questions about fulfilled general requirements and conditions

1. Has this research or similar research by the department been previously submitted to the EC?
   ☐ Yes,
   ☒ No
   If yes, what was the number allocated to it by the EC?
   Explanatory notes:

2. Under which category does the research fall with regard to the consideration of Medical / Not medical? (Also see Chapter 4.)
   - ☒ Category D
   - ☐ Category A
   - ☐ Category B
   - ☐ Category C
   - ☐ Uncertain, explain why

   Explanatory notes: *This is non-medical research with negligible risk, hence category D.*

3. Are adult, competent subjects selected?
   - ☒ Yes, indicate in which of the ways named in the general requirements and conditions this is so
   - ☐ No, explain
   - ☐ Uncertain, explain why

   Explanatory notes: *Yes, all subjects signing up need to provide a number of details about themselves,*
   including their age. If the prototype reports another (likely) age for any subject this subject is eliminated
   from the research (and informed)

4. Are the subjects completely free to participate in the research, and to withdraw from participation whenever they wish and for whatever reason?
   - ☒ Yes
   - ☐ No, explain why not
   - ☐ Uncertain, explain why

   Explanatory notes: *Yes, subjects sign themselves up so there is a full informed consent. Subjects can withdraw at any time.*

5. In the event that it may be necessary to screen experimental subjects in order to reduce the risks of adverse effects of the research: Will the subjects be screened?
   - ☒ Screening is not necessary, explain why not
   - ☐ Yes, explain how
   - ☐ No, explain why not
   - ☐ Uncertain, explain why

   Explanatory notes: *The risk for subjects is nil, hence no screening is required.*

6. Does the method used allow for the possibility of making an accidental diagnostic finding which the experimental subject should be informed about? (See general conditions.)
   - ☐ No, the method does not allow for this possibility
   - ☒ Yes, and the subject has given signed assent for the method to be used
   - ☐ Yes, but the subject has not given signed assent for the method to be used
   - ☐ Uncertain, explain why

   Explanatory notes: *Yes, the whole purpose of the prototype is to deduce information about the subject. This might yield unexpected results. This will be mentioned in conditions before signing-up.*

7. Are subjects briefed before participation and do they sign an informed consent beforehand in accordance with the general conditions?
   - ☒ Yes, attach the information brochure and the form to be signed
   - ☐ No, explain why not

☐ Uncertain, explain why

Explanatory notes:

8. Are the requirements with regard to anonymity and privacy satisfied as stipulated in § 5.2.7?
   ☒ Yes
   ☐ No, explain why not
   ☐ Uncertain, explain why

   Explanatory notes: *The researcher has filled in a PII form and all data gathered will be stored in a XML*
   Database running on the researchers laptop. The directory containing all data is encrypted using EncFS encrypted on a "paranoid" level. The encrypted data will be backed-up on a USB stick / CD-ROM at the office of the first supervisor.

9. If any deception should take place, does the procedure comply with the general terms and conditions (no deception regarding risks, accurate debriefing)?
   ☒ No deception takes place
   ☐ The deception which takes place complies fully with the conditions (explain)
   ☐ The deception which takes place does not comply with the conditions (explain)

   If deception does take place, attach the method of debriefing

   Explanatory notes: *No deception takes place.*

10. Is it possible that after the recruitment of experimental subjects, a substantial number will withdraw from participating because, for one reason or another, the research is unpleasant?
    ☒ No
    ☐ Yes, that is possible

    If yes, then attach the recruitment text paying close attention to what is stated about this in the protocol.

    Explanatory notes

3.3 Questions regarding specific types of standard research

Answer the following questions based on the department to which the research belongs.

11. Does the research fall ***entirely*** within one of the descriptions of standard research as set out in the described standard research of the department?
    ☐ Yes, go to question 12
    ☐ No, go to question 13
    ☒ Uncertain, explain what about, and go to question 13

    Explanatory notes:

12. If yes, what type of research is it? Give a more detailed specification of parts of the research which are not mentioned by name in this description (for example: What precisely are the stimuli? Or: What precisely is the task?)

13. If no, or if uncertain, give as complete a description as possible of the research. Refer where appropriate to the standard descriptions and indicate the differences with your research. In any case, all possible relevant data for an ethical consideration should be provided.

The experiment tries to find online manifestations of the subjects and tries to determine which one's
actually belong to the subject and which not. There is a full informed concent, so the main issues are

security of the gathered data and privacy.