D2.2.1
Multilingual, Ontology-Based IE from Stream Media - v1

Niraj Aswani (USFD), Mark A. Greenwood (USFD), Kalina Bontcheva (USFD), Leon Derczynski (USFD), Julián Moreno Schneider (DFKI), Hans-Ulrich Krieger (DFKI), Thierry Declerck (DFKI)

Abstract.
FP7-ICT Strategic Targeted Research Project (STREP) ICT-2011-287863 TrendMiner
Deliverable D2.2.1 (WP2)

This deliverable introduces the first prototype implementation of two ontology backed IE systems as they started to be used in TrendMiner. One prototype incorporates DBpedia into the extraction process and performs disambiguation in order to unambiguously associate annotations with URIs. The second prototype implements a polarity lexicon acquisition strategy and makes use of a company ontology in order to support the storage of instance information, and a first version of the TrendMiner opinion ontology – still to be integrated in the web service – is being tested in a stand alone version of the IE system.

Keyword list: information extraction, ontologies, lexicon acquisition, opinion mining

Project	TrendMiner No. 287863
Delivery Date	November 9, 2012
Contractual Date	October 31, 2012
Nature	Prototype
Reviewed By	N/A
Web links	`http://demos.gate.ac.uk/trendminer/rest/service/annotate`
Dissemination	PU

TrendMiner Consortium

This document is part of the TrendMiner research project (No. 287863), partially funded by the FP7-ICT Programme.

DFKI GmbH
Language Technology Lab
Stuhlsatzenhausweg 3
D-66123 Saarbrcken
Germany
Contact person: Thierry Declerck
E-mail: declerck@dfki.de

University of Southampton
Southampton SO17 1BJ
UK
Contact person: Mahensan Niranjan
E-mail: mn@ecs.soton.ac.uk

Internet Memory Research
45 ter rue de la Rvolution
F-93100 Montreuil
France
Contact person: France Lafarges
E-mail: contact@internetmemory.org

Eurokleis S.R.L.
Via Giorgio Baglivi, 3
Roma RM
00161 Italy
Contact person: Francesco Bellini
E-mail: info@eurokleis.com

University of Sheffield
Department of Computer Science
Regent Court, 211 Portobello St.
Sheffield S1 4DP
UK
Contact person: Kalina Bontcheva
E-mail: K.Bontcheva@dcs.shef.ac.uk

Ontotext AD
Polygraphia Office Center fl.4,
47A Tsarigradsko Shosse,
Sofia 1504, Bulgaria
Contact person: Atanas Kiryakov
E-mail: naso@sirma.bg

Sora Ogris and Hofinger GmbH
Linke Wienzeile 246
A-1150 Wien
Austria
Contact person: Christoph Hofinger
E-mail: ch@sora.at

Hardik Fintrade Pvt Ltd.
227, Shree Ram Cloth Market,
Opposite Manilal Mansion,
Revdi Bazar, Ahmedabad 380002
India
Contact person: Suresh Aswani
E-mail: m.aswani@hardikgroup.com

Changes

Version	Date	Author	Changes
1.0	15.10.2012	Mark A. Greenwood	initial draft
1.1	25.10.2012	Mark A. Greenwood	added work on polarity classification
1.2	08.11.2012	Mark A. Greenwood	added the contribution from DFKI
1.3	09.11.2012	Mark A. Greenwood	finalized for internal review
1.4	09.11.2012	Mark A. Greenwood	changes based on review feedback

Executive Summary

This deliverable describes the first prototypes for the ontology based IE, disambiguation, and polarity classification. A number of different approaches are outlined which will allow us to support different use-case requirements. Specifically we introduce prototypes which allow us to processes both English text as well as Spanish and Italian.

Contents

Chapter 1

Introduction

Identification of Named Entities (NE) such as people, organisations and locations is fundamental to semantic annotation and is the starting point of more advanced text mining algorithms. For instance, sentiment analysis is widely used in finance to extract the latest signals and events from news that could affect stock prices. However, before extracting company-related sentiment, it is necessary to identify the documents containing the corresponding and *unambiguous* company entities. In this deliverable we present prototypes of both named entity extraction and sentiment analysis of stream media.

Humans usually resolve ambiguities based on context and the first prototype, reported in Chapter 2, uses Linked Data for extending the already available context. This prototype combines a state-of-the-art named entity tool with novel Linked Data-based similarity measures allowing us to disambiguate entities against Wikipedia entries.

We also present work on reputation polarity classification. While similar to sentiment analysis the classification is subtly different in that classically negative sentiment can have a positive affect on a brand. This complication, which will be explained in more detail in Chapter 3 means that it is not possible to use existing sentiment analysis tools in an off-the-shelf fashion. In a parallel development, we implemented a first version of a service for the semi-automatic generation of a polarity lexicon from a larger corpus of short financial news tickers.

Finally we briefly present the first version of TrendMiner opinion ontology, which re-uses and extends the existing Marl ontology[1].

1.1 Relevance to TrendMiner

TrendMiner aims to work over large volumes of streaming media and end users require the text to be categorized in order to help them quickly understand events. Polarity clas-

[1] `http://www.gi2mo.org/marl/0.1/ns.html`

sification is one way in which streaming media can be classified, and the information extraction and disambiguation approaches reported can serve as clues to polarity.

The majority of the work reported in this deliverable is related to Tasks 2.1 (Multilingual knowledge and lexical acquisition for customizing Ontology Schema) and 2.3 (Multilingual, ontology-based IE from stream media: entities, events, sentiment, and trends), although Chapter 4 relates to Task 2.2 (Ontological modelling and reasoning with opinions, sentiment, provenance and trend information). As this is a prototype deliverable and does not focus on evaluation only a small amount of work related to Task 2.4 is included where we thought it appropriate.

1.1.1 Relevance to project objectives

The work reported in this deliverable provides the software processing required for both information extraction, disambiguation and polarity classification over streaming media.

1.1.2 Relation to other workpackages

Information extraction, disambiguation and polarity classification are requirements of the two use cases (WP6 and WP7) as well as being preparatory steps required for the summarization work being carried out in WP4.

Chapter 2

Ontology Based IE and Disambiguation

The goal of this prototype is to identify named entities in text and attach the correct DBpedia URI to each one of them[1] . For the former, we use the ANNIE Information Extraction [CMBT02] system from GATE [CMB+11]. It combines some small lists of names (e.g. days of the week, months) and rule-based grammars, to processes text and produce NE types such as *Organization*, *Location* and *Person*. ANNIE also resolves co-reference so that entities with the same meaning are linked. For example, *General Motors* and *GM* would be identified as referring to the same entity.

GATE's ontology-based gazetteer, namely the Large Knowledge Gazetteer (LKB), is used for entity linking. LKB performs lookup and assigns URIs to words/phrases in the text. For the purpose of our application, we match only against the values of the *rdf:label* and *foaf:name* properties, for all instances of the *dbpedia-ont:Person*, *dbpedia-ont:Organisation* and *dbpedia-ont:Place* classes.

Both ANNIE and the LKB can be used independently, however, while NE types generated by ANNIE miss the URI which is necessary to disambiguate them, LKB does not use any context, which results in generating many spurious entities. For example, each letter *B* is annotated as a possible mention of *dbpedia:B_%28Los_Angeles_Railway%29*, which refers to a line called *B* operated by Los Angeles Railway. We next describe the algorithm which filters out such noise, by consolidating the output of ANNIE and LKB, followed by a disambiguation step. A high-level pseudo code looks as follows:

```
1. Identify NEs (Location, Organisation and Person) using ANNIE
2. For each NE add URIs of matching instances from DBpedia
3. For each ambiguous NE calculate disambiguation scores
4. Remove all matches except the highest scoring one
```

The disambiguation algorithm uses context in which the particular entity appears and a weighted sum of the following three similarity metrics:

- *String similarity*: refers to the Levenshtein distance between the text string (such as *Paris*),

[1]The majority of this section is taken from a previous description of the work, see [DB12].

and the labels describing the entity URIs (for example, *Paris Hilton*, *Paris* and *Paris, Ontario*).

- *Structural similarity* is calculated based on whether the ambiguous NE has a relation with any other NE from the same sentence or document. For example, if the document mentions both *Paris* and *France*, then structural similarity indicates that *Paris* refers to the capital of France. All other entity URIs can be disregarded, based on the existing relationship between *dbpedia:Paris* and *dbpedia:France*.

- *Contextual similarity* is calculated based on the probability that two words have a similar meaning as in a large corpus (DBpedia abstracts in our case) they appear with a similar set of other words. To implement that we use the Random Indexing method [Sah05] and calculate similarity using the cosine function.

2.1 Initial Evaluation

Although this is a prototype deliverable we thought it worth including a brief overview of the RepLab 2012 [ACG+12] profiling task in which we evaluated an earlier version of this approach. For full details of our submission please refer to [GAB12].

Our disambiguation approach to relevance filtering achieved an accuracy of 0.52. Note that the "all relevant baseline" has an accuracy of 0.71 which shows the large bias within the evaluation set towards relevant tweets. A similar bias can also be found within the training data which results in only a small amount of non-relevant examples. This bias may be a general occurrence or may be due to the specific entities chosen for this evaluation, i.e. many of the entities do not actually require disambiguation.

Chapter 3

Reputation Polarity Classification

3.1 Classification of English Text

The standard GATE distribution provides a number of machine learning tools which can be used to perform text classification[1]. The prototype, described in this deliverable, for polarity classification uses the Batch Learning PR configured to perform k-Nearest Neighbours (k-NN) classification using an implementation from Weka[FHH+05]. Whilst space constraints preclude full details of the implementation (which can be found in [CMB+11]) the algorithm is configured to use the following features for learning: POS tags (both 1-gram and 2-gram), emoticons, hashtags and the language of the tweet. Whilst it may seem strange that the words themselves (or at least their root forms) were not used to train the classifier, experimentation showed that including them led to a drop in performance of up to 5%. The reason behind this rather odd result is as yet unclear but may be related to the small amount of training data that was available (and hence only a small number of words occurring frequently) at development time.

3.1.1 Initial Evaluation

As with the disambiguation prototype described in the previous chapter, we evaluated the reputation polarity classifier as part of RepLab 2012 profiling task. The approach achieved an accuracy of 0.41. In comparison an all positive run achieves an accuracy of 0.44 (all neutral: 0.33, all negative 0.23).

We believe that relatively low performance of the current prototype is due to two things: the small amount of training data used and the difference in language use when expressing opinions across entities of different types. This second problem is probably more relevant than the lack of training data. During development we tested the algorithms using k-fold cross validation in two ways. In both cases we used six folds. One approach used the training data from one entity as a fold, and in the other data from all six entities were randomly split equally between the six folds. The average accuracy of the two approaches showed a difference of around 25%, with better

[1] see `http://gate.ac.uk/userguide/chap:ml` for details

performance being achieved when the folds were generated randomly. We believe that this is due to the fact that none of the six entities in the training data overlap in the products or services they provide and as such the language used to talk about them differs greatly. The six entities were:

- **Alcatel**: a provider of backend communications equipment, usually sold to governments or telecommunication companies rather than end-users
- **Apple**: a seller of consumer electronic goods including computers, phones and MP3 players
- **Armani**: a high-end fashion label
- **Barclays**: a British multinational banking and financial services company
- **Lufthansa**: the largest airline in Europe
- **Marriott**: a large chain of hotels and leisure resorts

As you can imagine complaining about a late flight (Lufthansa) would use very different language to complaints about short battery life in a consumer electronics product (Apple). This suggests that when building a classifier the training data should contain tweets about a variety of different entities. Extra data (from both the RepLab evaluation set, and from manual annotation carried out during TrendMiner) should allow us to improve classification performance well above the currently reported level.

3.2 Classification of Non-English Text

In a second prototype, developed at DFKI, we describe a web service that includes both the semi-automatic acquisition of polarity bearing lexical items and the recognition and polarity marking of companies listed in the Madrid stock exchange, including the mention of industry activity fields. The latter information is part of a company ontology we derived semi-automatically from Web pages of stock exchanges (like the German Stock Exchange[2]), which are delivering continuous stream of information (at least during opening times of stock exchanges). Part of the work has been done for German and English already in the context of the Monnet project[3]. In TrendMiner we extended this to Italian[4] and Spanish [5] [6].

3.2.1 Semi-Automatic Acquisition of Polarity bearing Lexical Items in Input Text Stream

We use in this first step a very robust method. All lexical items associated with a positive or negative stock value development of a company is considered as either positive or negative. Going

[2] http://deutsche-boerse.com

[3] http://www.monnet-project.eu/

[4] http://www.borsaitaliana.it

[5] http://www.bolsamadrid.es

[6] Note that while Spanish is not an official supported language within TrendMiner, we took advantage of a visiting research assistant to develop a Spanish prototype

over a large corpus of such short financial ticker news, the algorithm checks the number of positive or negative values associated with each word. We have to stress here, that before associating candidate polarity values to lexical items, those have been processed by a Part-of-Speech tagger and grouped into chunks by grammars written in the NooJ framework (`www.nooj4nlp.net/`), which is now also integrated in the web service. In the case of a huge disparity of values (see for example the item "pérdidas" below in Figure 3.1, the lexical items are getting associated with the predominant value. The second round of the acquisition process checks then the syntactic context of the lexical items clearly marked as being either positive or negative. We are at the stage of testing our algorithm that fixes the polarity of terms used in the surrounding syntactic context of the uniquely marked lexical items.

3.2.2 Detection of Companies, their associated Polarities and related Activity Fields

Getting our basic information for building the ontology schema from stock exchange web pages allows us to also extract in a straightforward manner the name and legal status of companies. Also abbreviations for the naming are given. In this we can build a high accuracy gazetteer on the fly. We also extract the activity fields of companies and other related information, like the names of the leading personal, and some basic financial figures. Some of the financial figures are stable (yearly revenue), some are changing continuously (value of the stock during a stock exchange session). The input text stream is given by financial tickers, and news-wire articles (both partly also delivered by the stock exchange page, but we also access other financial information services). In our first implementation round, we concentrate on the names and the activity fields of the companies.

3.3 Complementarity with the Entity Disambiguation Approach by Sheffield

We would like to stress that the approach we described in this section is complementary with the entity linking and disambiguation approach described in the first section and implemented by Sheffield. In fact both approaches will be combined. The entity disambiguation uses for now ontological knowledge available in the Linked Open Data sphere, and it is used mainly for accurate semantic annotation. But we all know that DBpedia is containing more data (for now) using English labels for naming the knowledge objects, so that it worth to check the use of other sources, and in dependency of specific applications also generate new knowledge sources, like we did for Spanish and Italian. And the second approach is adding to the semantic annotation of text (Company name and Company activity fields) also polarity values, which are then stored in the TrendMiner knowledge base. As a next step, the company ontology and the opinion ontology will be made available in the same ontological search space, the OWLIM (`http://www.ontotext.com/owlim`) platform of the TrendMiner partner Ontotext, so that all partners will be able to query not only the Linked Open Data world, but also the TrendMiner specific ontologies.

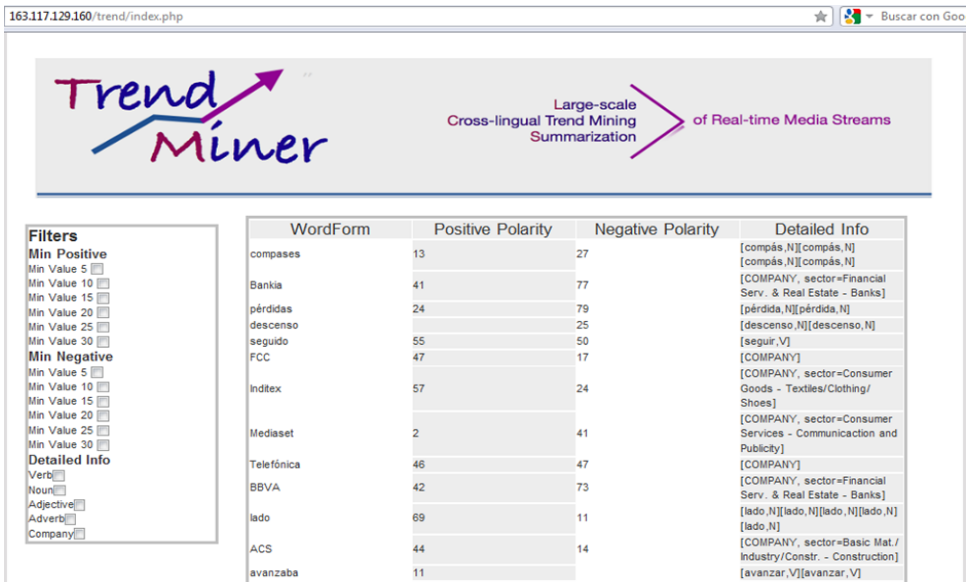

Figure 3.1: In this screenshot of the resulting page of our web service, the reader can see both the result of the automated extraction of polarity bearing words, and their rough distribution over positive and negative polarity (after round 1, applying the very basic algorithm). She can also see the result of information extraction, for now focused on company names and their activity sector (as extracted from the corresponding company ontology).

3.3.1 Initial Evaluation

It is certainly still too early in the project to present relevant evaluation studies. We just want to mention here the strategy we are implementing for the financial use case (in Italian). There we follow the same strategy for the acquisition of polarity bearing lexical and terminological items. We designed a form, which is representing the instance world of the opinion ontology, containing the name of the companies and the different associated opinion features we extracted. The use case partner, Eurokleis, has started then to manually validate the opinion value, to mark in the associated text segments the right words (if a correction of the output of the system is needed) that are leading to the opinion mark-up.

3.3.2 Next Steps

The web service will be extended to the full list of features associated to the company information we can extract from web pages and news tickers, and will be made available for German and Italian too. We note already that many processes of the actual approach are in fact language independent (robust acquisition of polarity items, generation of ontology schema, detection of company names).

Chapter 4

An Ontology For Opinions

4.1 Derivation of Ontology Schemas from Spanish and Italian Stock Exchange Web Pages.

A first step in our work consist in extracting from the Spanish (Madrid) and Italian (Milano) stock exchange pages the relevant information about companies, which should be later recognized in the analysis of text. This step is giving thus the ontology schema against which text analysis will be matched, and corresponding facts extracted from incoming text (streams of incoming financial ticker texts) can then be stored as instances in the TrendMiner knowledge base (the type of information we have been collecting for building the ontology schema is described in [DKG12]). All terms we extract from the web page (financial data like "Efectivo" – Turnover, or the activity fields: "Mineral, Metales y Transformación" – Mineral, Metal and Transformation) are stored in the language specific RDFs annotation properties (mainly in the label feature) associated with the ontology classes and properties. This gives us a part of the vocabulary to be used for recognizing relevant text segments in input text stream.

4.2 Establishing an Ontology Schema for Opinion

In order to be able to store the opinion value associated with detected companies in the input stream text, we decided to opt for an independent opinion module for opinion. The starting point of this ontology is given by the work of [WIR], resulting in the Marl Ontology[1]. Marl is an ontology designed to use for publishing the results of the opinion mining process in a form of structured data on the Web. The goal is to unify the access the numerical approximation of the opinion concept and allow extensive reasoning and search over multiple resources from different content providers. The DFKI Opinion Ontology re-uses, modifies and extends the original Marl Opinion Ontology. The object property `extractedFrom` has been redefined as a datatype property. Functionality has been imposed on certain properties. Range types have been defined for certain properties. Three new properties have been added: (i) `hasHolder`, (ii) `holdersTrust`, (iii)

[1] `http://marl.gi2mo.org/0.2/ns.html`

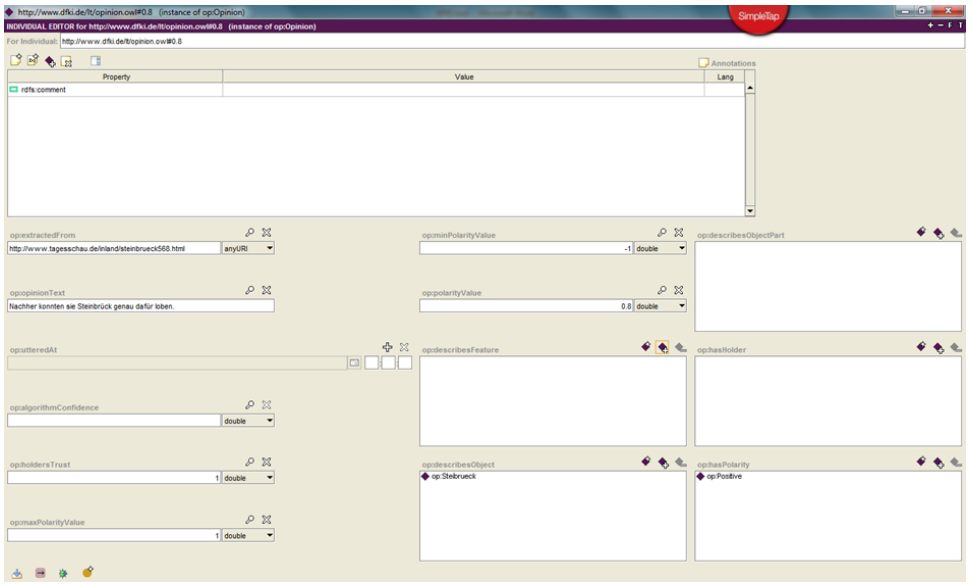

Figure 4.1: In this screenshot from the ”Individuals” panel of the Protégé tools, one can see the number of ”slots” reserved for an opinion stated towards the German politician Peer Steinbrück.

`utteredAt`. This ontology will be continuously updated. The instance world of the opinion ontology is storing each opinion, which is considered to be relevant to the TrendMiner use cases and partners. In this instance word, we can then store the name of the opinion holder (person, institution or poll etc.), the text source, the specific text segment(s), the polarity (negative, positive or neutral), the polarity value (in function of a minimum and maximum possible values, etc.). This instantiation world is displayed below in Figure4.1: For all other information, the interested reader can consult the Marl page.

Chapter 5

Software Availability

Disambiguation is available via a web service described at

`http://demos.gate.ac.uk/trendminer/lodie/`

This service uses the software described in this deliverable to process text to produce Mention annotations linked to DBpedia. To use the service POST an XML based request (the Content-Type paramter must be to set to `application/xml`), such as the following, to

`http://demos.gate.ac.uk/trendminer/lodie/service/annotate`

```
<request>
  <text>President Obama had flown back to United States after visiting Iran's president.</text>
</request>
```

A successful response to this request would be

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<message>
  <msg></msg>
  <status>SUCCESS</status>
  <text>
    <![CDATA[&lt;Mention inst="http://dbpedia.org/resource/Barack_Obama"&gt;President Obama&lt;/Mention&gt; had flown
         back to &lt;Mention inst="http://dbpedia.org/resource/United_States"&gt;United States&lt;/Mention&gt; after
         visiting &lt;Mention inst="http://dbpedia.org/resource/Iran"&gt;Iran&lt;/Mention&gt;'s president.]]>
  </text>
</message>
```

If an error occurs then the `status` element would read ERROR and the underlying exception would be available within the `msg` element of the response. Note that for performance reasons the service currently restricts the text to process to no more than 2000 characters.

A simple web based demo is also available at

`http://demos.gate.ac.uk/trendminer/obie/`

Bibliography

[ACG+12] Enrique Amigó, Adolfo Corujo, Julio Gonzalo, Edgar Meij, and Maarten de Rijke. Overview of RepLab 2012: Evaluating Online Reputation Management Systems. In *CLEF 2012 Labs and Workshop Notebook Papers*, 2012.

[CMB+11] H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan, N. Aswani, I. Roberts, G. Gorrell, A. Funk, A. Roberts, D. Damljanovic, T. Heitz, M.A. Greenwood, H. Saggion, J. Petrak, Y. Li, and W. Peters. *Text Processing with GATE (Version 6)*. 2011.

[CMBT02] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, 2002.

[DB12] Danica Damljanovic and Kalina Bontcheva. Named Entity Disambiguation using Linked Data. In *Proceedings of the 9th Extended Semantic Web Conference*, 2012.

[DKG12] Thierry Declerck, Hans-Ulrich Krieger, and Dagmar Gromann. Acquisition, Representation, and Extension of Multilingual Labels of Financial Ontologies. In Rute Costa, Manuel Silva, and António Lucas Soares, editors, *Proceedings of the TKE Workshop "Challenges to Knowledge Representation in Multilingual Contexts"*, pages 17–26. TKE, 6 2012.

[FHH+05] E. Frank, M.A. Hall, G. Holmes, R. Kirkby, B. Pfahringer, and I.H. Witten. Weka: A machine learning workbench for data mining. *Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers*, pages 1305–1314, 2005.

[GAB12] Mark A. Greenwood, Niraj Aswani, and Kalina Bontcheva. Reputation Profiling with GATE. In *CLEF 2012 Labs and Workshop Notebook Papers*, 2012.

[Sah05] M. Sahlgren. An introduction to random indexing. In *Proc. of the Methods and Applications of Semantic Indexing Workshop*, Copenhagen, Denmark, 2005.

[WIR] Adam Westerski, Carlos A. Iglesias, and Fernando Tapia Rico. Linked Opinions: Describing Sentiments on the Structured Web of Data. In *Proceedings of the 4th International Workshop "Social Data on the Web" (SDoW2011)*.