Assessment of Utility in Web Mining for the Domain of Public Health

Peter von Etter, Silja Huttunen, Arto Vihavainen, Matti Vuorinen and Roman Yangarber
University of Helsinki
Finland
First.Last@cs.helsinki.fi

Abstract
This paper presents ongoing work on application of Information Extraction (IE) technology to domain of Public Health, in a real-world scenario. A central issue in IE is the quality of the results. We present two novel points. First, we distinguish the criteria for quality: the objective criteria that measure correctness in traditional terms (F-measure, recall and precision), and on the other hand, subjective criteria that measure the utility of the results to the end-user.

Second, to obtain measures of utility, we build an environment that allows users to interact with the system by rating the analyzed content. We then build a classifier that learns from the user’s responses, to predict the relevance scores for new events. We conduct experiments with learning to predict relevance, and discuss the results and their implications for text mining in the domain of Public Health.

1 Introduction
We describe an on-going project for text mining in the domain of Public Health. The aim of the project is to build a system for providing decision support to Public Health (PH) officials and professionals the field of Epidemic Surveillance.

Epidemic surveillance may be sub-divided into indicator-based vs. event-based surveillance, (Hartley et al., 2010). Whereas the former is based on structured, quantitative data, which is collected, e.g., from national or international laboratories or databases, and is of reliable quality, the latter is much more noisy, and relies on alert and “rumour scanning”, particularly from open-source media, such as on-line news sources. While the latter kind of information sources are less reliable overall, they nonetheless constitute a crucial channel of information in PH. This is because the media are extremely adept at picking up isolated cases and weak signals—which may be indicative of emergence of important events, such as an incipient epidemic or change in a public-health situation—and in many cases they can do so much more swiftly than official channels. National and supra-national (e.g., European-level) Health Authorities require timely information about threats posed to the public by emerging infectious diseases and epidemics. Therefore, these Agencies rely on media-monitoring as a matter of routine, on a continual basis as part of their day-to-day operations.

Our system is designed to support Epidemic Surveillance by monitoring open-source media for reports about events of potential significance to Public Health (Yangarber and Steinberger, 2009). We focus in this paper on news articles mentioning incidents of infectious diseases. The system does not make decisions, but filters massive volumes of information and tries to identify those cases that should be brought to the attention of epidemic intelligence officers (EIO)—public health specialists engaged in epidemic surveillance.

This is an inter-disciplinary effort. The system builds on methods from text mining and computational linguistics to identify the items of potential interest(Grishman et al., 2003). The EIOs, on the other hand, are medical professionals, and are generally not trained in computational methods. Therefore the tools that they use must be intuitive and must not overwhelm the user with volume or complexity.

A convenient baseline is keyword-based search, as provided by search engines and news aggrega-
tors. Systems that rely on keyword-matching to find articles related to infectious threats and epidemics quickly overwhelm the user with a vast amount of news items, much of which is noise.

We have built a three-staged system for Epidemic Surveillance. First, the system tries to identify articles potentially relevant to Epidemic Surveillance, using a broad keyword-based Web search. Second, the system employs fact-finding technology, known as information extraction (IE), to determine exactly what happened in the article, who was affected by what disease/condition, where and when—creating a structured record that is stored in the database. Articles that do not trigger creation of a database record are discarded. A third component then determines the relevance of the selected articles—and cases that they describe—to the domain of Public Health.

Traditionally in IE research, performance has been measured in terms of correctness—how accurately the system is able to analyze the article (Hirschman, 1998). In this paper we argue the need for other measures of performance for text mining, using as a case study the application of Web mining to the domain of Public Health. In the next section, we lay down criteria for judging quality, and present the approach taken in our system. Section 3 outlines the organisation of the system, and Section 4 presents in detail our experiments with automatic assignment of relevance scores. In the final section we discuss the results and outline next steps.

2 Criteria for quality

In this section we take a critical view at traditional measures of quality, in text analysis in general, and IE in particular. What defines quality most appropriately for our application, and how should we measure quality? We propose the following taxonomy of quality in our context:

- Objective: system’s perspective
  - Correctness
  - Confidence
- Subjective: user’s perspective
  - Utility or relevance
  - Reliability

At the top level, we distinguish objective vs. subjective measures. Most IE research has focused on correctness over the last two decades, e.g., in the MUC and ACE initiatives (Hirschman, 1998; ACE, 2004). Correctness is a measure of how accurately the system extracts the semantics from an article of text, in terms of matching the system’s answers to a set of answers pre-defined by human annotators. In our context, a set of articles is annotated with a “gold-standard” set of database records, each record containing fields like: the name of the disease/infectious agent, the location/country of the incident, the date of the incident, the number of victims, whether they are human or animal, whether they survived, etc. Then the system’s response can be compared to the gold standard and correctness can be computed in terms of recall and precision, F-measure, accuracy, etc.—counting how many of the fields in each record were correctly extracted. This approach to quality is similar to the approach taken in other areas of computational linguistics: how many structures in the text were correctly identified, how many were missed, and how many spurious structures were introduced.

Confidence has been studied as well, to estimate the probability of the correctness of the system’s answer, in e.g., (Culotta and McCallum, 2004). Our system computes confidence using discourse-level cues, (citation withheld): e.g., confidence decreases as the distance between event trigger and event attributes increases—the sentence that mentions that someone has fallen ill or died is far from the mention of the disease. Confidence also depends on uniqueness of attributes—e.g., if a document mentions only one country, the system has more confidence that an event referring to this country is correct.

On the subjective side, utility, or relevance, asks how useful the result is to the user. There are several points to note. First, it is clearly a highly subjective measure, not easy to capture in exact terms. Second, it is “orthogonal” to correctness in the sense that from the user’s perspective utility matters irrespective of correctness. For example, an extracted case can be 100% correct, yet have very low utility to the user, (for the task of epidemic surveillance)—a perfectly extracted event that happened too long ago would not matter in the current context. Conversely, every slot in the record may be extracted...
erroneously, and yet the event may be of great importance and value to the user. We focus specifically on relevance vs. correctness.

Given the current performance “ceilings” of 70-80% F-measure in state-of-the-art IE, what does correctness of x mean in practice? It likely means that if \( x > y \) then a system achieving F-measure \( x \) is better to have than one achieving \( y \). But what does it say about utility? In the best case, correctness may be correlated with utility, in the worst case it is independent of utility (e.g., if the system happens to achieve high correctness on events from the past, which have low relevance). Since we are targeting a specific user base, the user’s perspective must be taken into account when estimating quality, not (only) the system’s perspective. This implies the need for automatic assignment of relevance scores to analyzed events or documents.

Finally, reliability measures whether the reported event is “true”. The relevance of extracted fact may be high, but is it credible? Can the information be trusted? We list this criterion for completeness, since it is the ultimate goal of any surveillance process. However, answering this requires a great deal of knowledge external to the system, that can only be obtained by the human user through a detailed downstream verification process. The system may provide some support for determining reliability, e.g., by tracking the performance of different information sources over time, since the reliability of the facts extracted from an article is related to the reliability of the source. It may be possible to classify Web-based sources according to their credibility; some sources may habitually withhold information (for fear of impact to tourism, trade, etc.); other sites may try to attract readership by exaggerated claims (e.g., tabloids). On the other hand, clearly disreputable sites may carry true information. This measure of quality is beyond the scope of this paper.

3 The System: Background

We now describe the surveillance system, some of whose components have been described previously, (citation withheld for anonymity). In several aspects, it is similar to other existing systems for automated epidemic surveillance, viz., BioCaster (Doan et al., 2008), MedISys and PULS (Yangarber and Steinberger, 2009), HealthMap (Freifeld et al., 2008), and others (Linge et al., 2009). Due to space limitations, we cannot provide a detailed description of the system here.

First, the system’s component for IR (information retrieval) performs a broad Web search, using a set of boolean keyword-based queries, (citation withheld). The result is a continuous stream of potentially relevant documents, updated every few minutes. Second, an IE component, (citation withheld), analyzes each retrieved document, to try to find events of potential relevance to Public Health. The system stores the structured information about every detected event into a database. The IE component uses a large set of linguistic patterns, which in turn depend on a large-scale public health ontology, similar to MeSH,\(^1\) that contains concepts for diseases and infectious agents, infectious vectors and animals, medical drugs, and geographic locations.

The result of IE is a populated database of extracted “facts”, that can be browsed and searched according to the user’s interests. It is crucial to note that the notion of a user’s focus or interest is not the same as the notion of relevance, above. We take the view that the notion of relevance is shared among the entire PH community: an event is either relevant to PH or it is not. Note also, that this view is upheld by several classic, human-moderated PH surveillance systems, such as ProMED-Mail\(^2\) or Canadian GPHIN. User’s interest is individual, e.g., a user may have specific geographic, or medical focus (e.g., only viral or tropical illnesses), and given the structured database, s/he can filter the content according to specific criteria. But that is independent of the shared notion of relevance to PH. User focus can be exploited for targeted recommendation, using techniques such as collaborative filtering; at present, this is beyond the scope of our work.

The crawler and IE components have been in operation and under refinement for some time. We next build a classifier to assign relevance scores to each extracted event and matched document.

\(^1\)www.nlm.nih.gov/mesh  
\(^2\)www.promedmail.org
4 Experimental Setup

We now present the work on automatic classification of relevance scores. In collaboration with the end-users, we defined guidelines for judging relevance on a 6-point scale, summarized in Table 1.

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>New information, highly relevant</td>
<td>5</td>
</tr>
<tr>
<td>Important updates, on-going developments</td>
<td>4</td>
</tr>
<tr>
<td>Review of current events, potential risk of disease</td>
<td>3</td>
</tr>
<tr>
<td>Historical/non-current, background</td>
<td>2</td>
</tr>
<tr>
<td>Non-specific, non-factive events, secondary topics</td>
<td>1</td>
</tr>
<tr>
<td>Unrelated to PH</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 1: Guidelines for relevance scores in medical news

It is clear that the guidelines are highly subjective, and cannot be encoded by rules directly. We extracted features—the discourse features—from the document that are indicative of, or mappable to, the relevance scores. Examples of discourse features are e.g., Relative-position, which is represented by a number from zero to 1 indicating the proportion of the document one needs to read to reach the event text; Disease-in-header is a binary value that indicates whether the disease is mentioned in the headline or the first two sentences; Disease-trigger-distance indicates how far is the disease from the trigger sentence (same as for confidence computation); Recency is the number of days between the reported occurrence of the event and the publication date; and so on. We compiled over two dozen discourse-level features. ((Huttunen et al., 2002)). It is also clear that the discourse features do not determine the scores, but provide weak indicators of relevance, so that probabilistic classification is necessary. For example, a higher relative position of an event probably indicates lower relevance, but there are often news summary articles that gather many unrelated news together, and may contain very important items anywhere in the article.\(^3\) A feature such as Victim-named, stating whether the victim’s name is mentioned, often indicates low-relevance (obituaries, stories about public personalities, etc.). However, sometimes news articles about disease outbreaks deliberately personify the victims, to give the reader a sense of their background, lifestyle, to speculate about the victims’ common circumstances.

We next describe two classifiers we have built for relevance. A Naive Bayes classifier (NB) was used as the baseline. We then tried to obtain improved performance with Support Vector Machines (SVM).

**Data:** The dataset is the database of facts extracted by the system. The system pre-assigns relevance to each event, and users have the option to accept or correct the system’s relevance score, through the User Interface, which also allows the users to correct erroneous fills, e.g., if a country, disease name, etc., was extracted incorrectly by the system.

Along with the users, members of the development team also looked at the extracted events, and corrected relevance and erroneous fills. The developers are computer scientists and linguists, whereas the users are medics, and because they interpreted the guidelines differently this had an impact on the results, described in Tables 2 and 5.

Because the users (and developers) corrected events (in the two rightmost columns in the tables), we obtained two parallel sets of examples with relevance labels: the raw examples, as extracted by the system, and “cleaned” examples, after users/developer corrections. The raw set is more noisy, since it contains errors introduced by the system. We used the cleaned examples to train our classifiers, and tested them on both the cleaned set and the raw set. Testing against the cleaned set gives an “idealized” performance, (as if the IE system made no errors in analysis). True performance is expected to be closer to testing on the raw set.

In total, there were just under 1000 examples labeled by the users and the developers (some examples were labeled by both, since the system allows multiple users to attach different relevance judgments to the same example. Most of the time users agreed on the relevance judgments, but non-developers were less likely to clean examples.)

**Naive Bayes classifier:** The plain NB classifier did not perform well, because the discourse features we chose are not independent, which affects the per-
formance of NB.\(^4\) We therefore built a binary classifier instead. This was sensible in the context of our system, which provides two views, a *front page view* that contains only highly-relevant information (rated 4 or 5), in case the user wants to see only the most urgent items first, and the *complete* view that shows the user all extracted information. To try to reduce the mutual dependence among the features, we added a simple, greedy feature-selection phase during training. Feature selection starts by training a classifier on the full set of features, using leave-one-out (LOO) cross-validation to estimate the classifier’s performance. In the next phase, the algorithm in turn excludes the features one by one, and runs the LOO cross-validation again, once with each feature excluded. The feature whose exclusion gives rise to the biggest increase in performance is dropped out, and the selection step is repeated with the reduced set of features. We continue to drop features until performance does not increase for several iterations which, in our experiments, is three steps beyond the top performance. We then back up to the step that yielded peak performance. The resulting subset of features is used to train the final NB classifier.

The NB classifier is implemented in R Language.

Because relevance prediction is difficult for all events, we also tried to predict the relevance of an article, making the simplifying assumption that the article is only as relevant as the first event found in the article.\(^5\) The results are presented in Table 2, where *Dev only* refers to the data labeled by developers, and *Users only* to that labeled by users.

The event-level classification is shown in the top portion of the table. Throughout, as expected, testing on the cleaned data usually gives slightly better (more idealized) performance estimates than testing on the raw. Also, as expected, testing on the first-only events (document-level) gives slightly better performance, since it’s a simpler problem—although there is less data to train/test on.

It is important to observe that using data labeled by developers gives significantly higher performance. This is because coercing the users to follow the guidelines strictly is not possible, and they deviate from the rules that they themselves helped articulate. The rows labeled “all” show performance when all available labeled data, developers and the users combined, was used.

This performance is quite good for a baseline.\(^6\) The confusion matrices—for the developer-only event-level raw data set—show the distribution of true/false positives/negatives.

<table>
<thead>
<tr>
<th>Predicted labels</th>
<th>True Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-5</td>
<td>125</td>
</tr>
<tr>
<td>0-3</td>
<td>42</td>
</tr>
</tbody>
</table>

Table 3: NB confusion matrix

**SVM Classifier:** For comparison, we implemented two extra classifiers using the SVMLight Toolkit\(^7\). We used a simple linear kernel as our baseline, and experimented with the potentially more expressive RBF kernel. The conditions for testing the SVM classifiers were same to the ones for NB, and same datasets were used as the baseline.

SVM with the RBF kernel can use non-linear separating hyperplanes in the original feature space by using the so called kernel trick (Aizerman et al., 1964). RBF kernel can be computed effectively and since it is not limited to simple linear decision

---

\(^4\)We originally planned to do regression to the [0–5] scale, but that proved problematic since the amount of data was not sufficient enough to cover the “grey area” between very relevant and not-so-relevant.

\(^5\)A manual check confirmed there where no instances where the first event in an article had lower relevance than a subsequent event.

\(^6\)Consider for comparison, that the correctness on a manually constructed, non-hidden set of articles used for system development, is under 75% F-measure.

\(^7\)http://svmlight.joachims.org/
boundaries we felt that it might improve the results over the baseline linear kernel. A more detailed discussion of SVM and different kernel functions for text classification is beyond the scope of this article. For more information see for example (Joachims, 1998).

To regularize the data for SVM, the feature values were normalized to lie between 0 and 1 (for continuous features), and set to 0 or 1 for binary features. Table 4 describes accuracy achieved with default linear kernel of SVMLight. All discourse features describe tests when using the whole discourse feature set (over 20 features). Selected discourse features present results from training with exactly same features as was used for NB after feature selection.

Table 4: SVM prediction accuracy using linear kernel

<table>
<thead>
<tr>
<th></th>
<th>Event-level</th>
<th>Document-level</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Clean</td>
<td>Raw</td>
</tr>
<tr>
<td>All discourse</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dev only</td>
<td>75.33</td>
<td><strong>77.17</strong></td>
</tr>
<tr>
<td>All</td>
<td>71.60</td>
<td>72.26</td>
</tr>
<tr>
<td>Selected discourse features only</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dev only</td>
<td>76.07</td>
<td><strong>77.95</strong></td>
</tr>
<tr>
<td>All</td>
<td>71.40</td>
<td>72.14</td>
</tr>
</tbody>
</table>

Table 5: SVM prediction accuracy using RBF kernel

The difference when training with selected discourse features and all discourse features is minor, as SVM is able to distinguish between relevant and non-relevant features. The results from SVM using linear kernel are comparable with the results from NB.

In addition to using the discourse features, we also tried using lexical features. The lexical features for a given example—extracted event—is simply the bag of words from the sentence containing the event, plus the two surrounding sentences. To reduce data sparsity, the sentences are pre-processed by a lemmatizer, and passed through a named entity (NE) recognizer (citation withheld), which replaces persons, organizations, locations and disease names with a special token indicating the NE’s class. “Stop-word” parts of speech were dropped—prepositions, conjunctions, and articles.

As the performance of RBF kernel is highly influenced by its parameters C (trade-off between training error and margin) and \( \gamma \) (kernel width) (Joachims, 1998), we performed simple manual tuning for the RBF parameters. Manual tuning was performed by building a grid of values, and finding areas where a testing dataset performed well. These areas were then further investigated. We tried total 40 different combinations and fixed C to 10000 and \( \gamma \) to 0.001 for evaluations. The results for SVM using RBF kernel are given in Table 5.

High performance of lexical features alone was surprising as lexical features consist only of the event sentence and sentences before and after the event sentence. News documents often have event information scattered around the document, we call this the shotgun effect. For example, location can be in the headline, disease in the middle of the document, and event sentence in a third location. ((Huttunen et al., 2002))Our lexical feature construction does not take this into account.

The difference between relevance in data labeled by developers and everyone comes mostly from the fact that developers follow given guidelines strictly, where “all” contains several different organizations, each following their own guidelines. There is also deviation within organizations. For example, certain doctors may find specific diseases or locations more interesting, giving events containing them a high relevance, thus involving personal preference with document relevance.
5 Discussion and Conclusions

SVM with manually tuned parameters performs better than RB by a small margin, though there is still much to be explored and improved. One odd effect is that sometimes testing on the raw data gives slightly better results than testing on the clean data, though this is probably not significant, since the SVM classifier is still not finely tuned (and the data contain some noise). Using all discourse features performs slightly worse than using a reduced set of features—the same set of features that we obtained through greedy feature selection for NB.

Although the lexical features alone seem to do somewhat worse than the discourse features alone on event-level classification, we still see that the lexical features contain a great deal of information (which the NB cannot use). As expected, adding the discourse features improves performance over lexical features alone, since discourse features capture information about long-range dependencies that local lexical features do not.

In forming splits for cross-validation or LOO, we made sure not to split examples from the same document across the training and test sets. That is, for a given document, all events in it are either used for training or for testing, to avoid biasing the testing.

To summarize, the points addressed in this paper:

- We have presented a language-technology-based approach to a problem in Public Health, specifically the problem of event-based epidemic surveillance through monitoring on-line media.

- The user’s perspective needs to be taken into account when estimating quality, not just the system’s perspective. Utility to the user is at least as important as (if not more important than) correctness.

- We have presented an operational system that suggests articles potentially relevant to the user, and assigns relevance scores to each extracted event.

- For now, we assume the users share same notion of relevance of an event to Public Health.

- We have presented experiments and an initial evaluation of assignment of relevance scores.

- Experiments indicate that relevance appears to be a tractable measure of quality, at least in principle. Marking document-level relevance—only for the first event in the document—appears to be easier. However, making real users follow strict guidelines is difficult in practice.

On-going work includes refining the classification approaches, especially, using Bayesian networks, regression, using transductive SVMs to leverage unlabeled data, and exploring collaborative filtering to address users’ individual interests.

References


