I, CODER –
EXAMINING THE IMPACT OF ARTIFICIAL INTELLIGENCE ON EVENT DATA CODING

John Wiesel
Free University of Berlin
SFB700: Governance in Areas of Limited Statehood
john.wiesel@fu-berlin.de

Christian Ickler
Free University of Berlin
SFB700: Governance in Areas of Limited Statehood
c.ickler@fu-berlin.de

Preliminary Draft
Please do not cite without the authors' permission.

Paper prepared for the 53rd Annual ISA Convention San Diego, 01-04 April 2012

Abstract

The internet promises ad hoc availability of any kind of information. Conflict researchers are seemingly only bound by the effort needed to find and extract the information from international news sources, which have become available at a fingertip. This begs the question whether the sheer number of accessible news sources and the speed of the news cycle dictate an automated coding approach in order to keep up? Will the initial costs of implementing such a system outweigh the possible loss of information? We answer these questions for the "Event Data on Conflict and Security" project (EDACS)[1] and carry out both human and machine assisted coding to generate temporal and spatial disaggregated event data for armed conflicts. In this pilot, we compare both approaches in a quantitative analysis and qualitatively by using spatial-temporal comparability measures. While the quality of human-coding exceeds a pure automated approach, a compromise between efficiency and quality results in a supervised semi-automated machine learning approach. We conclude by critically reflecting on the possible discrepancies in the analysis of these resulting datasets.

Keywords: Conflict Event Data, Machine Coding, Space-Time-Comparability, Data Quality.

1. Introduction

Spatially and temporally disaggregated event data has become the backbone of quantitative conflict science literature. A growing number of georeferenced datasets provides the necessary information on violent incidences in armed conflict (Chojnacki et al., 2012, Dulic, 2010, Melander and Sundberg, 2011, Raleigh et al., 2010). These spatiotemporal disaggregated datasets enable researchers to analyze variations of violence in time and space (cf. Buhaug, 2010, Raleigh, et al., 2010, Weidmann et al., 2010).

The Event Data on Conflict and Security project (EDACS) builds up and maintains one of these datasets. EDACS focuses on violence in areas of limited or failed statehood. The dataset enfolds seven countries of Sub-Saharan Africa (Burundi, Democratic Republic Congo, Liberia, Republic of the Congo, Rwanda, Sierra Leone, and Somalia) between 1990 and 2009. While this effort is currently being finalized, it took years and several thousands of working hours to complete the process. One of the most time consuming and error-prone phases of the data generation is the step of data transformation or "coding" (cf. Chojnacki, et al., 2012), which has been an almost entirely manual task. The rise in numbers and increased availability of news sources over the internet raises the question whether the sheer number of accessible news sources and the speed of the news cycle dictate an automated coding approach in order to keep up? Will the initial cost of implementing such a system outweigh the possible loss of information? Will such as system be able to achieve the necessary degree of data quality?

Based on these questions we carry out an experiment using a machine learning to generate spatial and temporal disaggregated event data of the armed conflict in Sierra Leone in the year 1999 and compare the resulting dataset with human-coded event data. In the first part we describe the Event Data on Conflict and Security project (EDACS) and refer to why computer-supported natural language processing is relevant to conflict event data projects and how we implemented our ML-based approach.

In the second part we will discuss this experiment, frame our comparative approach by discussing the experimental design, the costs originating from the two different approaches, the level of data comparability, and the overall pros and cons of machine- and human-coding. Furthermore we briefly present the methods to perform a stepwise comparison of machine learning and human-coded conflict event data along their spatial and temporal attributes. We therefore firstly analyze the time series in both datasets, in search for similar trends. Secondly we map and compare the spatial distribution of the two datasets and thirdly we apply spatio-

temporal-k-functions and plot matching events defined by a narrow spatiotemporal threshold, which is based on a SQL-query combined with the results of a spatiotemporal cluster analysis. In order to understand the task at hand, we begin by outlying the corner stone's of EDACS, its goal, scope, and core definitions.

2. Transformation of News Articles into Conflict Event Data

The transformation of natural language to structured event data requires a thorough conceptualization in order for the resulting database to achieve relevance. Any data project also has to know the inherent challenges of the data generation process. We will begin by explaining the concepts behind EDACS and other data projects, then touch on the subject of data quality in the field of conflict research and explain two crucial steps of the data generation process, the selection of source documents and subsequent extraction of events.

2.1 Conceptualization of Conflict Event Data

In EDACS, the basic unit of analysis is an event, defined as a violent incident with at least one fatality resulting from the direct use of armed force. Events are coded with their location (name of location, longitude and latitude coordinates) and timeframe (start and end date in case of events lasting more than one day). Among others, the type of military action (fighting[2] or diverse forms of one-sided violence[3]) is coded in EDACS, as well as the involved violent or non-violent actors and details on (civilian and military) fatalities. Events can be based on one or several news sources. For each event, the names and publication dates of all news articles used are indicated. Beyond that, EDACS-coders mark an article as *"biased"* if its information originates from a source directly connected to a violent actor involved in the respective event.

EDACS is built on information retrieved from newspaper articles. These articles are gathered from the Lexis-Nexis news portal. EDACS is based on a set of four predefined

sources or media outlets that are used for all coded countries and years of observation. The archives of three international newspapers (The Guardian, New York Times, and Washington Post) and the broad collection of translated local news reports by BBC Monitoring are searched by keywords through the news portal. In case of inconsistent information or missing data on one of EDACS central variables (location and timeframe of event), the four mandatory sources are supplemented by other sources such as other news services (trust.org/alertnet, irinnews.org, crisisgroup.org, humansecuritygateway.com), and regional internet gateways (allafrica.com, africa-confidential.com, reliefweb.int).

Each news article found by the search engine is read by EDACS-coders who extract the relevant information and enter it into a data entry form. In order to ensure intersubjectivity and data consistency, a set of strict and conservative coding rules has been developed. Additionally, all data is coded and double-checked by two different coders and cross-checked by a supervising coder (cf. Fig. 1 – Human Coding Workflow).

(Figure 1 about here)

Events are localized with longitude and latitude coordinates (WGS 84) using the toponymic GEOnet Names Server (GNS) provided and maintained by the US National Geospatial-Intelligence Agency (NGA, 2011).[4] GNS-data is an extensive settlement dataset, which is easily accessible at no charge. In case of imprecise ("between town A and town B") or ambiguous (duplicate location names in the GNS-data) indications of event locations in a primary sources, EDACS-coders consult additional map data such as GoogleEarth or Harvard's AfricaMap and/or apply a variety of standardized buffer rules.[5]

EDACS differs from other semi-automated georeferenced conflict event data projects, like the Armed Conflict Location and Events Dataset (ACLED), by its stricter event definition. Among others the number and type of fatalities as well as the involved violent or non-violent actors is coded in EDACS.[6] EDACS covers, unlike the Uppsala Conflict Data Program

Georeferenced Events Dataset (UCDP-GED), events with unknown actor participation and includes actors and dyads not surpassing the UCDP-GED-threshold of 25 conflict-related fatalities per year (Melander and Sundberg, 2011). Thereby EDACS provides on the one hand a more comprehensive view on patterns of violence in the observed countries factoring in all, also unidentified actors, but on the other hand UCDP-GED data may be more reliable because only events with clearly identifiable actors, that surpass the 25 fatality threshold, which might be more relevant to the armed conflict as such, are considered (Chojnacki, et al., 2012).

2.2 Challenges to Data Quality

Event data projects, like EDACS, face in general four categories of challenges to data quality: firstly errors and bias contained in the source (news) or secondly the auxiliary data (maps, etc.) and faults in the processes of transformation of source data into event data (misinterpretation, oversights, etc.) or thirdly in the contextualization of the events using auxiliary data (event localization, actor identification, etc.) (Chojnacki, et al., 2012).

While machine coding techniques could, in theory, be used to address each of these challenges, we are evaluating how automated source selection and data transformation impacts data quality. When measuring data quality, we will consider the aspects: completeness, accuracy, consistency, and relevancy (Batini and Scannapieca, 2006: 40, Thion-Goasdoué et al., 2007).

For instance, machine coding can improve speed of the coding process and thereby increase the completeness of the resulting data. Machine coding may be susceptible to errors contained in the auxiliary data and possible faults in the processes of spatiotemporal contextualization. Fuzzy specification of locations in the sources can deteriorate the performance of any geocoding approach (Pasley et al., 2007) just as ambiguous location names (ambiguous toponyms) can (Clough, 2005, Leidner, 2007).

Furthermore, while machine coding aims at improving on the accuracy and consistency of the transformation of raw text into structured data, results can be very misleading. Already in 2003, King and Lowe experiment with machine coding of events, or event extraction using the proprietary software provided by Virtual Research Associates, Inc. called VRA-Reader (King and Lowe, 2003). Although "virtually identical" accuracy of machine coding to human coding was reported, upon closer examination, the results were very mixed: a high precision of 93% was accompanied by also a false positive rate of 77%, i.e. 77% of sources were wrongly classified as containing an event (King and Lowe, 2003: 632). Using an unfiltered corpus of documents, their approach would result in vast amounts of false data and would eventually render any automated coding result useless if applied to other event data projects. A further downside to using proprietary software for this purpose is that it may decrease openness of the resulting database and deteriorate reliability, as argued by (Kauffmann, 2008: 108).

Today, almost all data projects such as ACLED, Minorities at Risk, or UCDP either rely on manually coded data, or use simple lists and lookup mechanisms for data generation (Nardulli et al., 2011: 10). But adaptive ML techniques have shown to outperform static natural language processing (NLP) approaches in particular when facing "noisy" data such as news reports in the conflict domain (Carlson et al., 2009, Sarwagi, 2007). Due to this finding and the progress in NLP, some event data projects such as the Integrated Crisis Early Warning System (ICEWS) (Schrodt, 2011) and the Social Political and Economic Events Database project (SPEED) project (Nardulli, et al., 2011) are currently making the transition to adaptive NLP-based event extraction techniques. While ICEWS has not yet documented any results of their transition from the static, rule-based NLP software Textual Analysis By Augmented Replacement Instructions (TABARI) to a new system, SPEED has already implemented a system to identify possibly relevant articles and uses NLP to find proper nouns in text to help

coders identify participating actors or location names. An adaptive, custom trained, NLP addition to the system is under development. In this case study for EDACS, we have implemented and completed a similar system that goes this one step further and makes use of artificial intelligence to learn how to identify and distinguish between actors, casualties and locations, and directly annotate news articles. To prevent the system from generating too many false positives and reduce overall workload, it is necessary to preselect or filter the relevant from the irrelevant source documents.

2.3 Filtering the Filters

In the context of conflict research, it can be argued that to achieve a certain level of representativeness is equivalent to a high degree of completeness or relevancy: “a sample of even 5% of [real] events would not be problematic if it were truly representative” (Earl et al., 2004). Thus to have a variety of databases of the same subject of investigation is a clear advantage, as it enables us to compare analysis across datasets (Chojnacki, et al., 2012). We will therefore compare between two different sets of sources:

The first is retrieved by a simple keyword search of the LexisNexis archive and restricted to four different media outlets (The Guardian, The New York Times, and The Washington Post and the broad collection of translated local news reports by BBC Monitoring). The sources are then manually processed in their entirety and in chronological order.

Secondly, we repeat the retrieval but lift the restriction on the media outlets. From this roughly five times larger array of sources, we use a machine learning (ML) approach to select a sample of the sources for event extraction. We expect that both approaches’ results share some similarity if either method achieves to generate a database of a certain degree of completeness and relevancy.

The second approach aims to classify the sources and choose which documents are relevant to the subject of interest. In contrast to projects like UCDP who use a static approach

of VRA (Gleditsch et al., 2002, Harbom and Wallensteen, 2009) we employ an active, adaptive approach, that learns over time to select the right documents to the user. We aim to drastically reduce the number of documents that have to be reviewed manually. Adaptive document classification has become common and can be found in everyday email clients but has, only recently, also been applied in this field by (Nardulli, et al., 2011). Instead of finding events, the aim is to select, or classify, the documents that are either unrelated or relevant to the conflict to allow researchers to understand the dynamics of the armed conflict in Sierra Leone. A subset of these articles contains the conflict events. In order to do so, we pair two different models, naive Bayes and boosted decision trees to perform the candidate selection.[7] Classifiers based on ML algorithms need so-called "features" as input, normally a preselection of words, grammatical attributes or similar. As feature selection in text classification tasks often delivers varying outcomes (Kim et al., 2006: 1460), we employ a straightforward bag-of-words approach, without filtering out of common words (stop-word approach) or linguistic transformation such as stemming. Initially, as there is no trained model yet, random articles are selected from the documents. Beginning with the second coding session, models are trained from the documents that are discarded or used by the users at each start. The next possible candidate for event extraction is then selected from these documents.

2.4 Event Extraction

We complement the automated document classification with a method for ML based event extraction. In the section above we explained how, on a document level, classifiers are trained to select documents of interest from a large set of source documents, our corpus. In addition to this, we use ML on an event level, and use sequence labeling to identify phrases that signify an event according to the EDACS definition and the related entities such as actors, time and place.

Requirement for an event in the definition of the EDACS project are casualties from a violent event. Inspired by (Banko et al., 2008) we create a sequence tagger that performs as a casualty extractor and identifies phrases in which casualties are mentioned. The underlying model is based on conditional random fields as the model has proven to be highly efficient, although the calculation needed is central processing unit (CPU) intensive.[8] After each coding session, a new, updated model is trained incorporating the new training data into the model (see Fig. 1 – Machine Assisted Coding).

We employ the same approach to the identification of phrases denoting actors, locations and dates. While there may be proper nouns, there are also composite names, such as *"Liberian rebels"* or more general locations such as *"the border"*. The proper nouns may be found using existing, pre-trained taggers, but the common nouns are normally not found as they are usually not part of the training data used to train the model. Furthermore, they are application domain specific. In a further step to support the annotation process, we combine both: a pre-trained tagger to identify actors and location names in text and a custom tagger that is trained using the annotations provided by the pre-trained tagger combined with the manually revised annotations to recognize these specific forms of actors and location names. Inspired by Stanford's approach to sequence tagging for named entity recognition (Finkel et al., 2005) and the conclusions drawn for our casualty extractor we use LingPipe's implementation for the custom tagger and pair it with Stanford's tagger, whose pre-compiled model achieved f-scores[9] of 86% on the 2003 corpus used at the Conference on Computational Natural Language Learning (Finkel, 2007, Finkel, et al., 2005). While sequence taggers can in principal be used to identify relationships occurring in sequences of words (Banko, et al., 2008), in this context, related entities do not necessarily appear in the same sequence, but may be sentences apart. When manually extracting events, relations become implicit when the user enters event by event. When annotating text, we have to explicitly define relations. We com-

bine the sequence labeling approach outlined above with an idea by (Ahn, 2006) and use the *"anchor"* phrases provided by the casualty tagger for ML-supported relation extraction.

Drawing upon all the entities and phrases identified above, the relation extractor classifies the relationships between these annotations. As per design, a casualty phrase forms the anchor of the event. This reduces the complexity of the task by evaluating all relationships between the entities in a text and a particular casualty phrase instead of all possible combinations. The relations are extracted as a binary classification task, using a maximum entropy classifier. An actor, a location or a date is either related to the event in question, or not.

2.5 Exemplary Workflow

First, the system iterates through all articles until one of the document classifiers identifies a candidate article. The program automatically executes the sequence taggers to identify possible casualty phrases, and entities such as locations and dates in the text. The example text in figure 2 contains three incidents.

The first is an attack by rebels of the Revolutionary United Front (RUF). The casualties, our event anchor, are identified correctly. It therefore also meets the minimum criterion of one casualty and is a valid target for event extraction. Two other incidents are mentioned, the second is the beginning of an offensive by forces of the Economic Community of West-African States Monitoring Group (ECOMOG) and the third inauguration of a new Chief of National Security. Both are not related to the first event and not events themselves according to the definition used here and therefore not to be extracted.

Second, the user reviews the tags presented and corrects all annotations that relate to the event accordingly. This is necessary, as even well trained annotators have shown to introduce a "very high" number of spurious instances (Giuliano et al., 2007). This could adversely affect any further steps, in our case the following relationship classification. The text highlights are updated in a different highlight style accordingly.

Third, the user runs the relation classifier to determine which annotations are related. Lastly, the user reviews if the relation classifier performs its task correctly. In the given example, it performs without error. If necessary, the user adds or removes relations where appropriate using a context menu. After finishing an article, the program selects the next candidate article for annotation. All annotations are stored by a open-source software library, Apache's Unstructured Information Management Architecture (UIMA) that uses the open standard XML (The Apache Software Foundation, 2010).

(Figure 2 about here)

After this data transformation of free text to semi-structured data we use a simplified, automated version of our data contextualization procedure. We automatically geocode location names (toponyms) and set coordinates by performing a lookup in the GNS-database (see Fig. 1 – Machine Assisted Coding). The software also deduces dates from temporal descriptions and meta-information such as the publication date using a small set of rules. The results are used as an ad-hoc set of data for comparison with our set of reference data, a subset of our manually generated and twice reviewed EDACS database.

3. Evaluation

Event extraction based on ML has already proven to be a useful application of artificial intelligence. The central question is, whether the effort to apply this technique to the domain of conflict research can be justified. We answer this question in the context of the EDACS project by a simple experiment that enables us to compare the events with the EDACS-dataset, which has been extracted, geocoded and proofed twofold – all manually.

First, we evaluate the performance of our approach in several dimensions: we look at the document classification approach by recoding the numbers of correctly identified relevant and irrelevant articles per session. Next, we evaluate the performance of the overall system by recoding the time needed to extract the resulting set of events. Finally, we investigate the resulting data quality more closely by measuring temporal and spatial similarity of the two datasets. But firstly, we will outline the approach of our experiment.

3.1 Experimental Setup

For the purposes of this experiment, we restrict the scope of our sources to the year 1999 and extract only events for the case of Sierra Leone. For the machine learning approach we retrieve 7,000 articles from all English language sources in the LexisNexis-archive for Sierra Leone 1999, based on a simple keyword search. For the manual approach, EDACS originally retrieved 1,200 articles from the four media outlets. As there is no training data in the beginning, i.e. extracted events, the software can only rely on pre-existing models that allow highlighting of actor names, locations and dates, but the software cannot yet identify common nouns as actor names, casualties, and relations, nor distinguish between relevant and irrelevant documents. Each session provides new training data, which the machine learning algorithm uses to create a mathematical model that is used by the software to present the user with the next candidate document, highlighting the identified phrases of relevance.

The participating coders are experts in manual coding but completely unfamiliar with machine-assisted coding and receive minimal training beforehand. In order to be spatially comparable to the human-coded EDACS-data we ask one of them to review the automatically assigned coordinates, as the geographic database used contains ambiguous entries. We also restrict the comparison to events with exactly specified dates and settlements only.

3.2 Cost Evaluation

One crucial, limiting element for all research is the available budgeting. The cost generated within projects can truncate primal objectives strongly and especially data projects depend – in particular whilst their set-up phase – on a sufficient budget. The budget limits the targeted coding project dimensions (number of coded cases and years, etc.) or possible retrospective refinements of project design and coding rules.

We exemplify the actual cost of coding for the case of Sierra Leone 1999 – comparing approximate human-coding costs with the costs of machine learning. We chose Sierra Leone 1999 because it equates an *"average"* swaying conflict year, with phases of escalation and de-escalation.

The cost originated from the two different approaches can be divided into four major categories: Facilities, Development, Coders, and Output (resp. number of events over time).The facilities and development cover: building occupancy expenses, office equipment, purchase of computers and software, providing a database server (for remote coding and centralized access); setup of a database, programming of a data entry form, etc., and development of the overall coding rules – what we subsume under facilities and development – is hard to quantify. The costs for facilities provided partly by the Collaborate Research Center 700 and the EDACS project amount roughly to 58,500 USD. The money and time spent on setting up the database and the first version of the data entry form only total to about 2123 USD. Further

development costs, including the salary of at least two research fellows (for five years) increase the expenses to about 320,579 USD.

In the case of machine learning, most of these costs also accumulate, but the crucial difference certainly is the specialized knowledge to implement such a system. There are no out-of-the-box solutions for ML-based event extraction which makes a custom development necessary. In this case, the actual software development of our prototype alone took up to half a year, while to develop a full-fledged "classic" database and entry form software system took roughly 160 hours. Overhead such as planning, research etc. is not included in these figures.

The costs of coding mostly depend on the research assistants' salary (in the case of EDACS 14.39 USD/hour), which equals about 120 read news article pages per hour; for Sierra Leone 1999, consisting of 1,442 pages for the four sources used within EDACS, this makes up to 172.92 USD. The ML approach in comparison only requires about 66 percent the working hours, and thereby is roughly 115.28 USD cheaper. One has to bear in mind, that every coded year has its own characteristics with regard to conflict dynamics and news coverage. Therefore the number of events per source fluctuates immensely. Still, the experiment spans about only 0.8 percent of the entire source data for the EDACS project. In a rough total, the ML approach could save more than 14,000 USD or 974 hours of manual work.

The generated output resp. the number of coded events per hour is unequal, whereas the human coders needed 12 hours for accomplishing the first round of coding, the prototype ML-coding only required two-thirds of the time. But this prototypical comparison is only feasible because the datasets contain differing details. The human-coded dataset offers more information but also requires a second round of coding. The lack in detail is further mitigated by the fact that the prototype ML approach does not identify duplicate events, every event is automatically annotated and then manually revised; the gross increase in event coded per hour of the ML approach compared to manual event coding was 156 percent.

3.3 Performance Evaluation

A prerequisite for efficient event extraction is to reduce source corpora to manageable sizes. Restricting the source articles to four media outlets and those with certain keywords accomplishes this task barely. About 3.2 percent of the articles are used by EDACS for event extraction in the case of Sierra Leone. Over 95 percent, thousands of articles per country have to be scanned and discarded manually. A fully automated event extraction approach used on an unfiltered corpus would lead to large amounts of false positives. By introducing a document classifier the number of documents used for extraction is reduced significantly. For the year 1999, a year with relatively intense fighting, about 90 percent of articles had to be discarded manually in the manual coding. When using the document classifier approach, on average, about 40 percent of articles selected by the classifier were deemed relevant by the human coder and half of those contained an event. Although we used a simple and fast classification method, and applied it to a corpus seven times as large as during our manual efforts, the ratio of events per document doubled on average. This decreases the overall workload as fewer documents have to be scanned manually to reach the same number of events. Figure 3 shows how the classifier improves over time beginning with session one; after a random subset of documents is manually processed and used as initial training data. Shown is the ratio how documents deemed relevant to all documents presented to the coder, in comparison to the manual coding displayed here as an average. During the manual extraction of events, in average 89 percent of the documents are discarded (the lower, continuous line). Due to the low amount of training data the algorithm achieved only 17 percent at first, but improved significantly over time, averaging 43 percent overall.

(Figure 3 about here)

Similar to the document classifier, event extraction started in session zero at minimal capacity. Random articles were reviewed and coded into events, if appropriate. On average, only three events were coded per hours, similar to the manual method. Already in the next session, where document classification preselected news articles and the taggers highlighted actors, locations, dates and casualties, the number increased to 8.4 events per hour. The coders averaged 9.4 events per hour. Accounting for duplicates, 5.2 unique events were coded per hour. This is an increase of 50 percent in comparison to the 3.67 unique events a trained and experienced coder codes manually. Pictured below is the performance of the human – machine tandem, a learning curve clearly visible.

(Figure 4 about here)

In summary, the overall throughput has greatly increased. The gross increase is 156 percent. If the system also achieves similar quality to the manual approach has yet to be determined. We will analyze and compare the data from a temporal, a spatial and a joint spatiotemporal perspective to determine whether there are similar trends and distributions.

3.4 Data Quality Evaluation

Spatiotemporal precision is the key aspect for any quantitative conflict analysis based on georeferenced conflict event data. There is no gold standard of conflict event data we can refer to, which is why we evaluate the spatiotemporal precision of the machine learning dataset, and thereby its data quality, in relative terms by comparing it to the manually generated EDACS dataset. Below we measure the similarity of both datasets temporally, spatially, and spatiotemporally.

3.4.1 Temporal Comparison

The coding of the exact date of a violent event is a challenging task, due to the fact that in 53 percent of human-coded events no exact date is provided by the source itself and only approx-

imate information is available (e.g. “two weeks ago”, “over the last few days”, …). In addition to imprecise temporal information regarding the circumstance of events, a substantial number of events are reported as aggregates (e.g. “over the course of the last two weeks”) and some sources give no temporal information at all. For comparative reasons we exclude the aggregated[10] events or events without any clear indicated date.

To measure temporal (dis-)similarity quantitatively the violent events are - in the further course - aggregated into weekly sums and in a first step, charted in a time series graph for descriptive analysis. In a subsequent analytical step we calculate the Granger-causality and the cross-correlation between the two time series.

(Figure 5 about here)[11]

The overall frequency, the number of events detected per time window seems, except for two substantive peaks in January and May 1999, mostly unison (see Fig. 5). Both peaks must be understood in the context of the historical events of the Sierra Leone Civil War. The escalation of violence at the turn of the years is rooted in a push of the rebels to retake Freetown, in January 1999 they overran most of the city whereas, and the drop to zero events per week in May can be attributed to the ceasefire agreement between the forces of President Kabbah and the Revolutionary United Front that took effect on May 24th and finally led to the Lomé Peace Accord (United Nations, 1999, United Nations, 2000).

(Figure 6 about here)

In order to control for trending in the data we detrend the data and perform seasonal adjustments. The detrended data, shown in figure 6, points to the fact that both datasets capture the main conflict developments, but appear to have slightly varying characteristics. On basis of the detrended data we calculate the Cross-Correlation-Function estimation (ccf[12]). The ccf-time series analysis shows positive, significant values. Especially the zero lag does signifi-

cantly correlate (see Fig. 7), that allows the conclusion that there is a strong temporal resemblance.

(Figure 7 about here)

The results of a Granger-causality-test of the two weekly aggregated, detrended and seasonality adjusted datasets show a highly correlated reciprocal effect (human-coded > machine learning [0.0582*]; machine learning > human-coded [0.0101**]). The Granger-causality suggests the finding that an increasing number of coded events at time *t(-1)*, in both datasets, positively affect the number of events at time *t(0)* (Granger, 1969). This also supports the view that both datasets do comprise a similar temporal trend.

3.4.2 Spatial Distribution

The spatial distribution of machine learning and human-coded data is utilized to evaluate their similarity purely within the spatial dimension. The overall spatial distribution (see Fig. 8) underlines the first impression gained by the temporal comparison and also seems to resemble the general course of events outlined above. The events of both datasets concentrate in the western part of Sierra Leone, but there are at least two distinct locations in each of the datasets which deviate from this pattern. Near Magburka are human-coded events present but machine learning events missing and in Kenema it is the other way around.

(Figure 8 about here)

Additionally to the cartographic mapping we also run Ripley's K clustering for spatial processes. The Ripley estimator summarizes spatial dependence (clustering or dispersion) over a range of distances and displays changes of the spatial dependence with regard to neighborhood size. Therefore the average number of neighboring events throughout the study area, evaluated with regard to their specific distance to one another, is compared to each events neighborhood and either considered clustered or disperse (ESRI, 2011). We are simulating

outer boundary values to correct for boundary effects, which can lead to an underestimation due to the number of neighbors for features near the edges of the study area of Sierra Leone (the simulated points are the duplicated points near the edges), and calculating 99 permutations for the confidence envelopes (ESRI, 2011).

First differences between the spatial clustering characteristics machine learning and human-coded data becomes visible. The machine learning data clusters gradually, decreasing with the increase in distance, whereas the human-coded event data clusters stronger locally, declines and finally levels after about 37 km distance (cf. Bivand and Gebhardt, 2000, Ripley, 1976, Rowlingson and Diggle, 1993). This suggests that the machine learning data is less clustered - especially on the local level - than the human-coded event data. The reason for that might be, beside a higher degree of spatial dispersion of the machine learning events, their smaller number.

3.4.3 Spatiotemporal Distance

Both spatial and temporal analysis of the conflict event data gathered can only provide a partial picture of the actual data resemblance. Therefore we finally use three different spatiotemporal analysis approaches to evaluate the overlap between machine learning and human-coded data: Firstly by comparison of the spatiotemporal K-function; secondly by spatiotemporal permutation Scan-statistics, and thirdly via SQL-based spatiotemporal similarity matching queries.

3.4.4 Spatiotemporal K-function

We start with the space-time K-function (stK). The space-time K-function estimates the extent of space-time clustering as a function of spatial and temporal separation based on second-order properties of a general stationary, homogeneous spatial-temporal Poisson point process. The space-time K-function is closely related to the Knox's statistic and tests the null hypothesis of no spatial and temporal interaction. Basis for the test is the theoretical intensity of the

expected number of events per spatial location and time unit and the observed number of points within a space-time cylinder centered on the event (for further information see: Cressie and Wikle, 2011: 210, Diggle et al., 1995: 125ff., Gabriel and Diggle, 2009: 45).

(Figure 9 about here)

The space-time interaction of the two datasets shows a high degree of similarity (see Fig. 9), whereas for the purely spatial cluster analysis (see above: 3.4.2 Spatial Distribution), the discrepancy between machine learning and human-coded data is partially bigger (Bailey and Gatrell, 1995, Diggle, et al., 1995, Rowlingson and Diggle, 1993).

3.4.5 Spatiotemporal Permutation Scan-statistics

The K-function only provides a global measure of spatiotemporal similarity. This is why we run further local spatiotemporal cluster analysis in order to identify similar clustering in both datasets. These matching spatiotemporal clusters are statistically significant data-specific hotspots of violence, which again, indicate similar trends - irrespective to the data source and data gathering technique used. The spatiotemporal permutation Scan-statistics provides values of local clustering[13] of violent events. The test statistic - and determining the cluster - is performed with the software SaTScan™. SaTScan creates a grid of centroids for the region and an infinite number of cylinders around each event location. The circular or ellipsoid radius of the cylinder reflects the portion of the events covered by the cluster; by default this does not exceed 50 percent of the total number of events. The height of the cylinder reflects time.

SaTScan calculates the likelihood function, obtaining actual and expected number of events, considering all events within the cylinders, testing for the Complete Spatiotemporal Randomness (CSTR).[14] Thereby the “most likely clusters” are identified and 999 Monte Carlo simulations are run, ranking/comparing the most likely clusters with randomly generated data via a Likelihood-ratio test. The Monte Carlo permutation procedure generates simulated da-

tasets and envelopes of 95% confidence interval for assessing the significance of the spatio-temporal permutation statistics. In a final step p-values for each cluster are calculated (Kulldorff and Information Management Services Inc., 2009, Kulldorff et al., 2005).[15]

The result of the spatiotemporal permutation SaTScan-statistics is sobering. There is only one significant cluster within both datasets detectable (see Fig. 10). A reason for this result can be the small sample size. The available number of coded violent events is problematic with regard to the reliability of the presented test statistics. As a rule of thumb there should be at least 30 events included into the calculations (here: 59 resp. 43 events). This raises the question of the robustness of the results, because the statistical results given here can only serve descriptive purposes and thereby only reveal approximate tendencies within the event data.

3.5.6 SQL-based Spatiotemporal Similarity Matching

We conclude the spatiotemporal comparison by plotting matching results for illustrative reasons on the map of Sierra Leone. This allows complement the cluster analysis, which did not allow robust statements, on the actual local comparativeness of the datasets. We match the datasets with the help of an SQL-query based upon defined spatial and temporal thresholds to cover, not only statistically significant, but all similarity and do not dependent on sample size. We run the SQL-query successively with an increment of one day and five kilometers, what led to the optimal Euclidian-distance-threshold of 20 km and a time window of 2 days.

(Figure 10 about here)

This narrow threshold reduces the number of matching events from 59 resp. 43 events to 23. The results show similarities but also substantial differences between conflict event data produced by the two presented approaches. The location and extent of the computed spatiotemporal SaTScan-clusters is also mapped in figure 10, as we can see the little resemblance be-

tween the two datasets captured by the cluster analysis - whereas the total number of SQL-based matching gives a very different picture. Notice that in figure 10 there are neither machine learning nor any human-coded space-time cluster around Port Loko, although the map shows a visual cluster based on the SQL-query-based matching events. The SQL-based similarity matching yields a spatiotemporal intersection of 38.98 resp. 53.49 percent of the coded events. This again emphasizes the result of almost all temporal and spatial evaluations measures above, that the conflict data events generated manually and via ML are very much alike.

Firstly the trends generated from the weekly aggregated, detrended and seasonally adjusted datasets seem to be similar. Secondly, the results of a cross-correlation analysis and the computed Granger-causality-test are in line with this view. The purely spatial comparison suggests that both datasets show related violent events, but with a few exceptions. Statistically valid answers to this guesswork delivered by global and local spatiotemporal cluster analysis corroborate this view. To meet robustness concerns due to the small sample size, the complimentary SQL-based spatial-temporal comparison broadens the basis of the assessment and leads to the final conclusion that machine learning and human-coding produces - with restrictions - similar events.

4. Discussion and Conclusion

The creation of spatiotemporal disaggregated conflict event data opens up possibilities for unpacking parts of the black box of war, and to get a more detailed view on conflict dynamics, actor constellations and thereby the processual nature of armed conflicts. This led to the growing importance of event data analysis in peace and conflict research that relies on precise and reliable data. Likewise the number of news sources and the speed of the information flow via modern ICT - even in remote areas and areas isolated by war - are rapidly increasing. We

propose to face this challenge by implementing a semi-automated machine learning event extraction approach.

The main goal for machine learning is to increase the throughput of event extraction. An important accompanying effect is a possible increase in openness and flexibility. We implemented an infrastructure that is based on open standards and stores all sources and their complete annotations which, copyright restrictions disregarded, allows for complete source transparency and makes ad hoc recoding possible at the same time. The entire method uses freely available libraries that create an adaptive approach to information extraction that can be applied to topics beyond conflict data generation as well.

During our evaluation, it turned out that the costs of implementation are marginal in comparison to the huge amount of time and money necessary to manually create a high quality conflict database such as EDACS. The increase in flexibility and throughput clearly outweighs the costs of human-coding. The ML based approach increased the number of events generated by 50 percent when accounting for duplicates. The gross increase was 156 percent. When extrapolated to the entire first coding rounds of EDACS even a 50 percent increase would have saved about 1,000 hours of manual coding. An enhanced ML approach that automatically detects duplicates would have saved more than 2,000 hours, about one year worth of manual work and equivalent of financial resources.

A necessary condition for the applicability of this method in the context of research is to achieve a high degree of reliability. We evaluated this key element of data quality by performing in-depth robustness check using manually generated data and assessed the spatiotemporal comparability of the machine learning dataset in contrast to the EDACS dataset which showed a high degree of similarity. The conducted trinity of temporal, spatial, and spatiotemporal comparison points to the fact that the machine learning dataset mirrors - to a large extent - the human-coded event data.

The discrepancies, which became apparent in the analysis, are marginal with respect to the spatiotemporal precision and revealed conflict trends. Still the comparability of other variables is yet unknown: we artificially restricted the comparison to key variables, information such as the aggregation of events, “fuzzy” event location or bias by the reporting agencies are all documented in the EDACS dataset but were not part of the evaluated ML data.

We are confident that creating a holistic ML approach for event extraction, that incorporates these facts, is feasible and a necessary step to achieve a higher degree of quality of event data. But it would seem ill-advised to neglect the human aspect of machine-assisted coding procedures. The minimal training our coders received was sufficient for the narrow scope of this experiment. To achieve even better data quality, good training is necessary. Ideally, well trained coders should be teamed up with a well adapted ML system that supports coders along each step with proposals for geocoding and ad hoc geo-information, context information such as conflict timelines, and real time duplicate detection to generate the best conflict event data possible.

These advances in computer sciences enable us to push the envelope of what is possible. They can and should be upscaled to more languages and be applied to other sources as well, as it is the only possibility to actually quantitatively prove the reliability of current conflict event data. Possible approaches include mining of crowd-sourced (e.g. crisismappers.net) or crowd-seeded data (e.g. Voix de Kivu).[16] Both approaches will have to answer data quality challenges outlined in this paper: crowd-sourcing’s participatory approach will have to tackle reliability issues as oversight is not inherent to the approach and robustness of reports have to be established. Crowd-seeding faces high initial cost as intensive planning and distribution is necessary to ensure that a representative sample of sources has been selected. Until next generation conflict event data projects are implemented on a significant scale, current event data

projects can adopt state-of-the-art technique as proposed in this paper to improve on their existing qualities.

References

Ahn, David. (2006) The Stages of Event Extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pp. 1-8. Sydney, Australia: Association for Computational Linguistics.

Bailey, Trevor, C., and Tony C. Gatrell. (1995) *Interactive Spatial Data Analysis*. 1 ed. Harlow, England: Longman Group Limited.

Banko, Michele, Oren Etzioni, and Turing Center. (2008) The Tradeoffs between Open and Traditional Relation Extraction. In *Proceedings of ACL-08: HLT*, pp. 28-36. Columbus, Ohio: Association for Computational Linguistics.

Batini, Carlo, and Monica Scannapieca. (2006) *Data Quality: Concepts, Methodologies and Techniques (Data-Centric Systems and Applications)*. Secaucus, NJ: Springer-Verlag New York, Inc.

Bivand, Roger, and Albrecht Gebhardt. (2000) Implementing Functions for Spatial Statistical Analysis Using the R Language. *Journal of Geographical Systems* 2:307-17.

Buhaug, Halvard. (2010) Dude, Wheres My Conflict? Lsg, Relative Strength, and the Location of Civil War. *Conflict Management and Peace Science* 27:107-28.

Carlson, Andrew, Scott Gaffney, and Flavian Vasile. (2009) Learning a Named Entity Tagger from Gazetteers with the Partial Perceptron.

Carpenter, Bob. (2010) Lingpipe 4.0.1. Alias-i.

Chojnacki, Sven, Christian Ickler, Michael Spies, and and John Wiesel. (2012) Event Data on Armed Conflict and Security: New Perspectives, Old Challenges, and Some Solutions. *International Interactions* forthcoming.

Cleveland, William S. (1981) Lowess: A Program for Smoothing Scatterplots by Robust Locally Weighted Regression. *The American Statistician* 35.

Clough, Paul. (2005) Extracting Metadata for Spatially-Aware Information Retrieval on the Internet. In *GIR '05 Workshop on Geographic Information Retrieval*, pp. 25-30: ACM.

Cohen, William W. (2004) Minorthird: Methods for Identifying Names and Ontological Relations in Text Using Heuristics for Inducing Regularities from Data.

Cox, David Roxbee, and Valerie Isham. (1980) *Point Processes*. Monographs on Statistics & Applied Probability: Chapman & Hall (CRC Press).

Cressie, Noel, and Christopher K. Wikle. (2011) *Statistics for Spatio-Temporal Data*. Wiley Series in Probability and Statistics. Hoboken, New Jersey: John Wiley & Sons, Inc.

Diggle, Peter. (2003) *Statistical Analysis of Spatial Points Patterns*. 2 ed. New York: Oxford University Press.

Diggle, Peter J., A. G. Chetwynd, R. Häggkvist, and S. E. Morris. (1995) Second-Order Analysis of Space-Time Clustering. *Statistical Methods in Medical Research* 4:124-36.

Dulic, Tomislav. (2010) Geocoding Bosnian Violence: A Note on Methodological Possibilities and Constraints in the Production and Analysis of Geocoded Event Data. In *Annual meeting of the Theory vs. Policy? Connecting Scholars and Practitioners.* New Orleans.

Earl, Jennifer, Andrew Martin, John D. McCarthy, and Sarah A. Soule. (2004) The Use of Newspaper Data in the Study of Collective Action. *Annual Review of Sociology* 30:65-80.

EDACS. (2011) Event Data on Conflict and Security (Edacs): Codebook. Version 3.1. Berlin.

ESRI. (2011) Arcgis Desktop: Release 10. Redlands, CA: Environmental Systems Research Institute.

Finkel, Jenny Rose. (2007) Named Entity Recognition and the Stanford Ner Software.

Finkel, Jenny Rose, Trond Grenager, and Christopher D. Manning. (2005) Incorporating Non-Local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, p. 363.

Gabriel, Edith, and Peter J. Diggle. (2009) Second-Order Analysis of Inhomogeneous Spatio-Temporal Point Process Data. *Statistica Neerlandica* 63:43-51.

Giuliano, Claudio, Alberto Lavelli, and Lorenza Romano. (2007) Relation Extraction and the Influence of Automatic Named-Entity Recognition. *ACM Transactions on Speech and Language Processing* 5:1-26.

Gleditsch, Nils Petter, Peter Wallensteen, Mikael Eriksson, Margareta Sollenberg, and Håvard Strand. (2002) Armed Conflict 1946–2001: A New Dataset. *Journal of Peace Research* 39:615-37.

Granger, Clive William John. (1969) Investigating Causal Relations by Econometric Models and Cross-Spectral Methods. *Econometrica* 37:424-38.

Harbom, Lotta, and Peter Wallensteen. (2009) Armed Conflicts, 1946-2008. *Journal of Peace Research* 46:577-87.

Kauffmann, Mayeul. (2008) Enhancing Openness and Reliability in Conflict Dataset Creation. In *Building and Using Datasets on Armed Conflicts*, edited by Mayeul Kauffmann. Amsterdam, Netherlands: IOS Press.

Kim, Sang-Bum, Kyoung-Soo Han, Hae-Chang Rim, and Sung Hyon Myaeng. (2006) Some Effective Techniques for Naive Bayes Text Classification. *IEEE Transactions on Knowledge and Data Engineering* 18:1457-66.

King, Gary, and Will Lowe. (2003) An Automated Information Extraction Tool for International Conflict Data with Performance as Good as Human Coders: A Rare Events Evaluation Design. *International Organization* 57:617–42.

Kulldorff, M., and Information Management Services Inc. (2009) Satscan^{tm} V9.0: Software for the Spatial and Space-Time Scan Statistics.

Kulldorff, Martin, Rick Heffernan, Jessica Hartman, Renato Assunção, and Farzad Mostashari. (2005) A Space-Time Permutation Scan Statistic for the Early Detection of Disease Outbreaks. *Public Library of Science (PLoS) Medicine* 2:216-24.

Leidner, Jochen Lothar. (2007) Toponym Resolution in Text Annotation, Evaluation and Applications of Spatial Grounding of Place Names. Institute for Communicating and Collaborative Systems School of Informatics University of Edinburgh.

Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schütze. (2008) *Introduction to Information Retrieval*. Cambridge: Cambridge University Press.

Melander, Erik, and Ralph Sundberg. (2011) Climate Change, Environmental Stress, and Violent Conflict - Tests Introducing the Ucdp Georeferenced Event Dataset. In *Annual meeting of th International Studies Association*. Quebec, Canada.

Nardulli, Peter F., Kalev H. Leetaru, and Matthew J. Hayes. (2011) Event Data, Civil Unrest and the Social, Political and Economic Event Database (Speed) Project: Post World War Ii Trends in Political Protests and Violence. In *Annual meeting of the International Studies Association*. Quebec, Canada.

NGA. (2011) Geonet Names Server (Gns). edited by National Geospatial-Intelligence Agency.

Norström, Madelaine, Dirk U. Pfeiffer, and Jorun Jarp. (2000) A Space–Time Cluster Investigation of an Outbreak of Acute Respiratory Disease in Norwegian Cattle Herds. *Preventive Veterinary Medicine* 47:107-19.

Pasley, Robert C., Paul D. Clough, and Mark Sanderson. (2007) Geo-Tagging for Imprecise Regions of Different Sizes. In *Proceedings of the 4th ACM workshop on Geographical information retrieval*, pp. 77-82. Lisbon, Portugal: ACM.

Raleigh, Clionadh, Andrew Linke, Håvard Hegre, and Joakim Karlsen. (2010) Introducing Acled: An Armed Conflict Location and Event Dataset. *Journal of Peace Research* 47:651-60.

Ripley, Brian D. (1976) The Second-Order Analysis of Stationary Point Processes. *Journal of Applied Probability* 13:255-66.

Rowlingson, Barry S., and Peter J. Diggle. (1993) Splancs: Spatial Point Pattern Analysis Code in S-Plus. *Computers in Geosciences* 19:627-55.

Sarwagi, Sunita. (2007) Information Extraction. *Foundations and Trends in Databases* 1:261-377.

Schrodt, Philip A. (2011) Automated Production of High-Volume, near-Real-Time Political Event Data. In *New Methodologies and Their Applications in Comparative Politics and International Relations*. Princeton.

The Apache Software Foundation. (2010) Apache Uima.

Thion-Goasdoué, Nugier Virginie, Sylvaine,, Dominique Duquennoy, and Brigitte Laboisse. (2007) An Evaluation Framework for Data Quality Tools. In *International Conference for Information Quality (ICIQ)*, pp. 280-94. Cambridge, MA.

United Nations. (1999) Secretary-General Welcomes Ceasefire Agreement on Sierra Leone.

———. (2000) Sierra Leone - Unomsil: Background. edited by Information Technology Section/ Department of Public Information (DPI): Peace and Security Section of DPI in cooperation with the Department of Peacekeeping Operations.

Venables, William N., and Brian D. Ripley. (2002) *Modern Applied Statistics with S*. Springer-Verlag.

Weidmann, Nils B., Jan Ketil Rød, and Lars-Erik Cederman. (2010) Representing Ethnic Groups in Space: A New Dataset. *Journal of Peace Research* 47:491-99.

Appendix

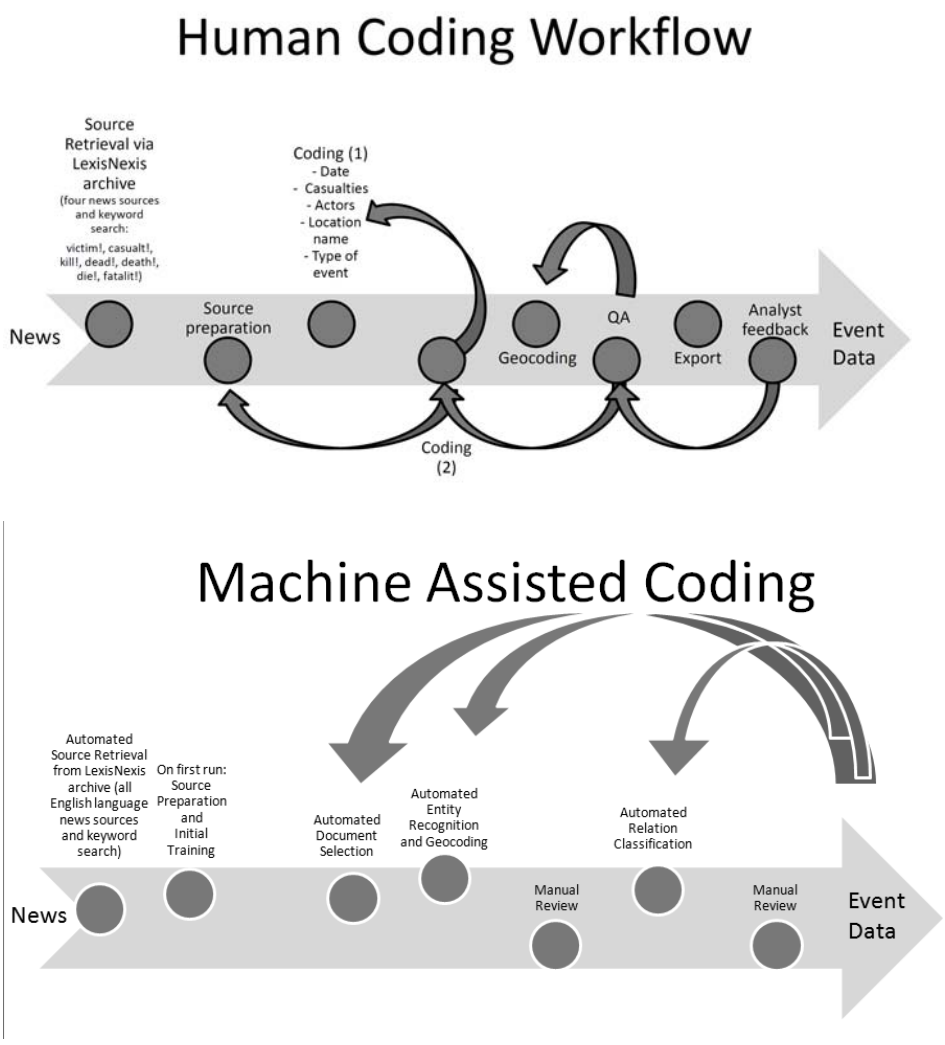

FIGURE 1 Human-Coding Workflow and Process of Machine Assisted Coding.

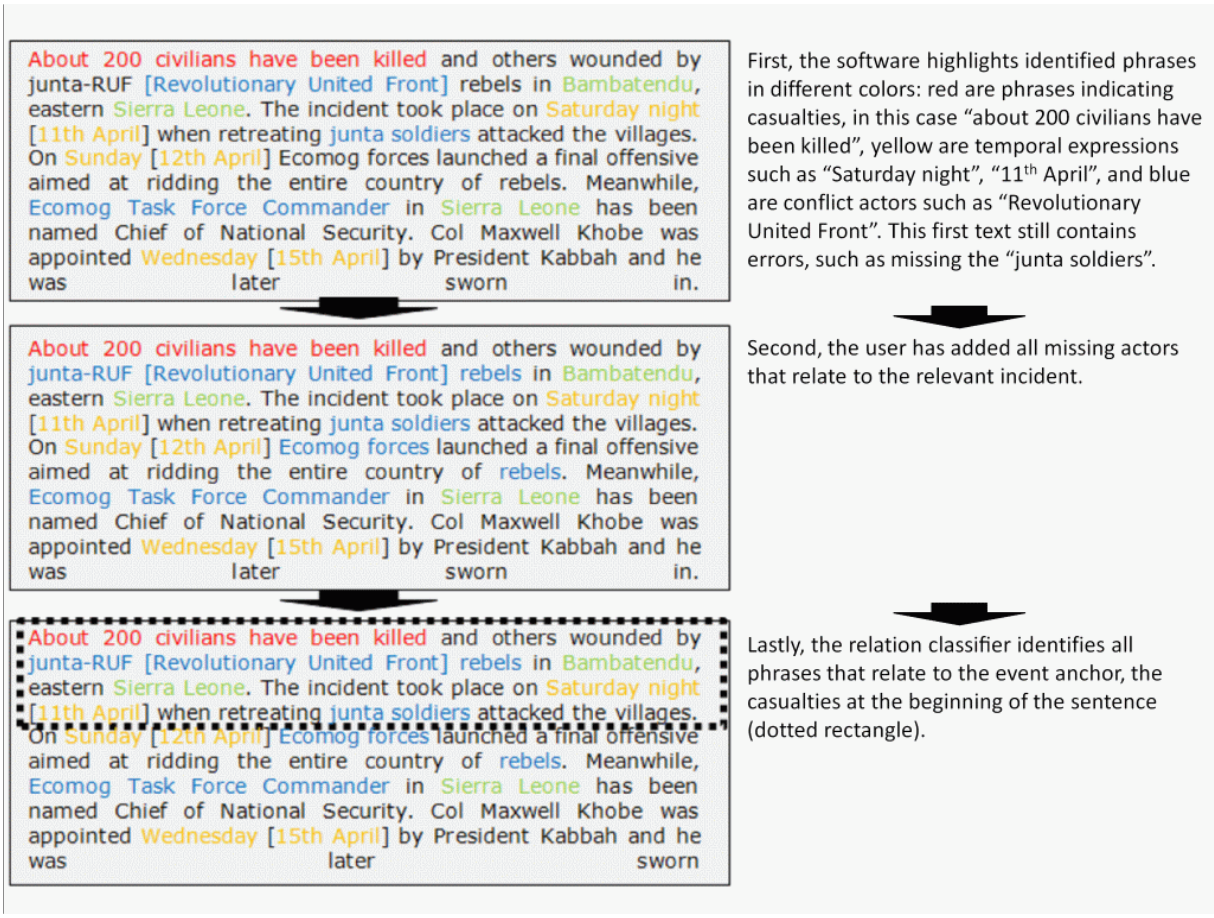

FIGURE 2 The machine-assisted coding consists of three steps. First, the software highlights identified phrases in different colors, second, the user has added all missing actors that relate to the relevant incident, and lastly, the relation classifier identifies all phrases that relate to the casualties at the beginning of the sentence (dotted rectangle).

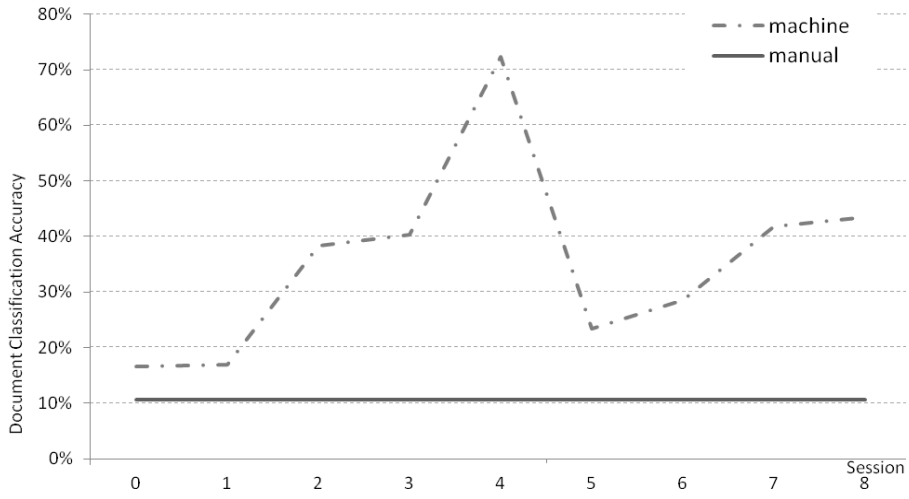

FIGURE 3 Document classification accuracy over time, beginning at session one. On average 43 percent were deemed as relevant by human coders in comparison to the about 10 percent baseline**.**

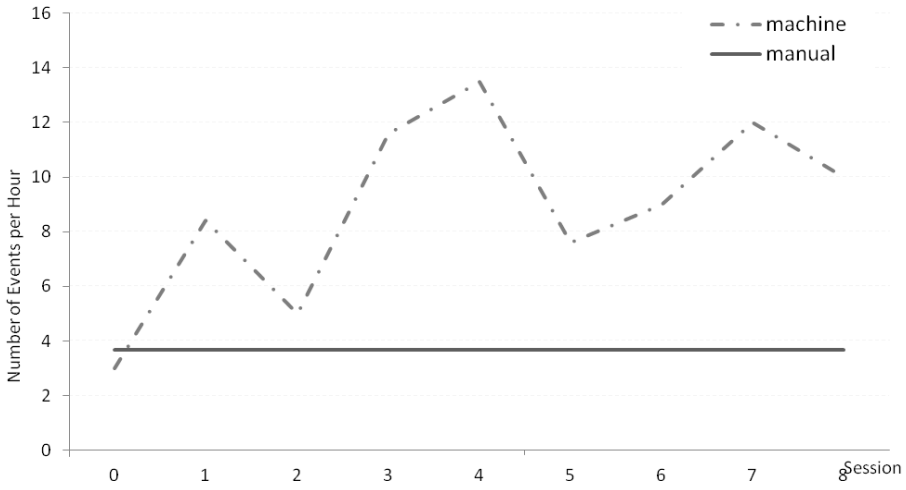

FIGURE 4 The number of machine-assisted extracted events per hour and session is shown, in comparison the baseline, the average manually coded events per hour.

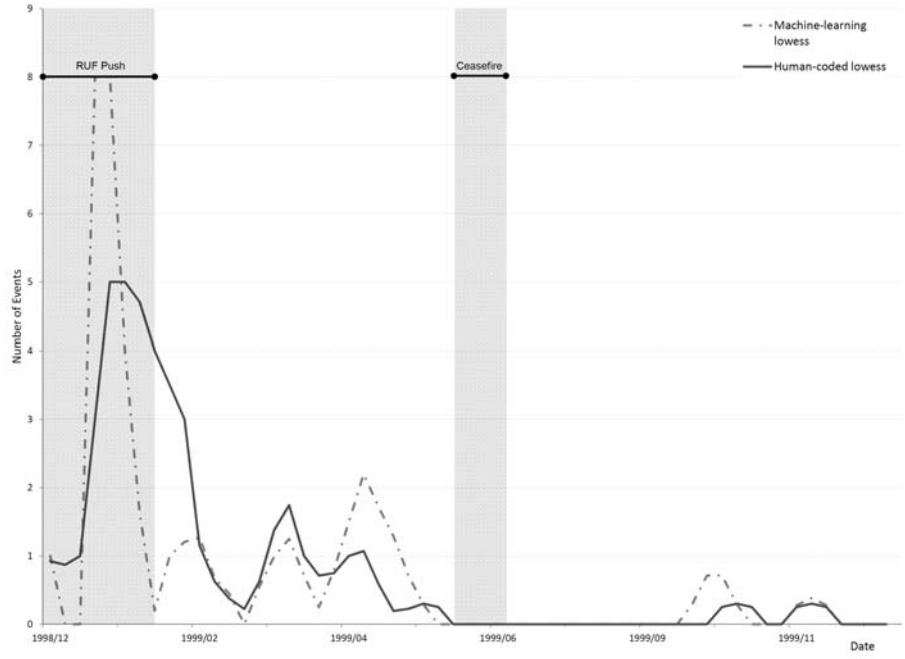

FIGURE 5 Weekly Time Series of Events of Machine Learning- and Human-Coded EDACS Data for Sierra Leone 1999.

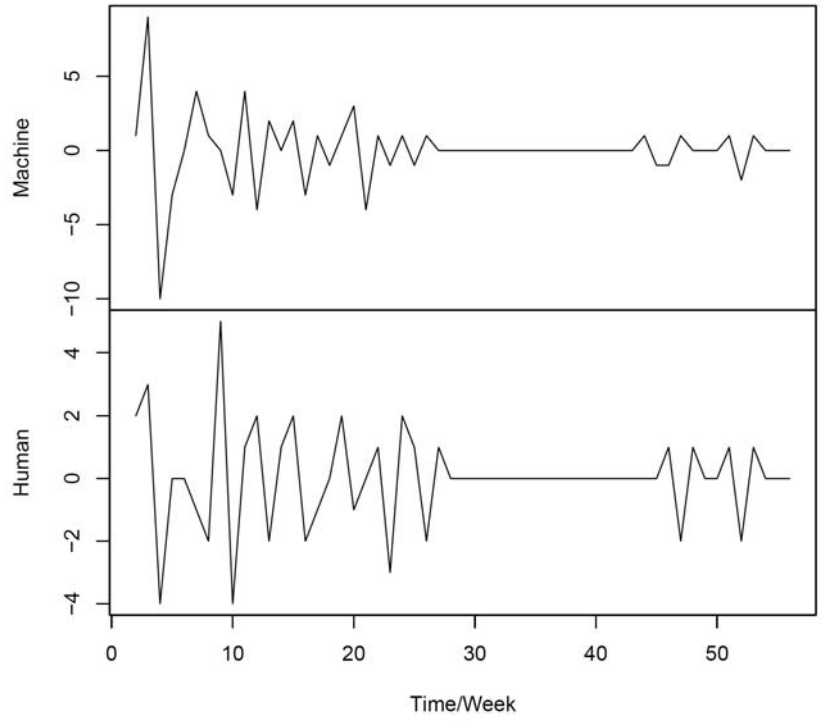

FIGURE 6 Detrended Time Series of Events of Machine Learning- and Human-Coded EDACS Data for Sierra Leone 1999.

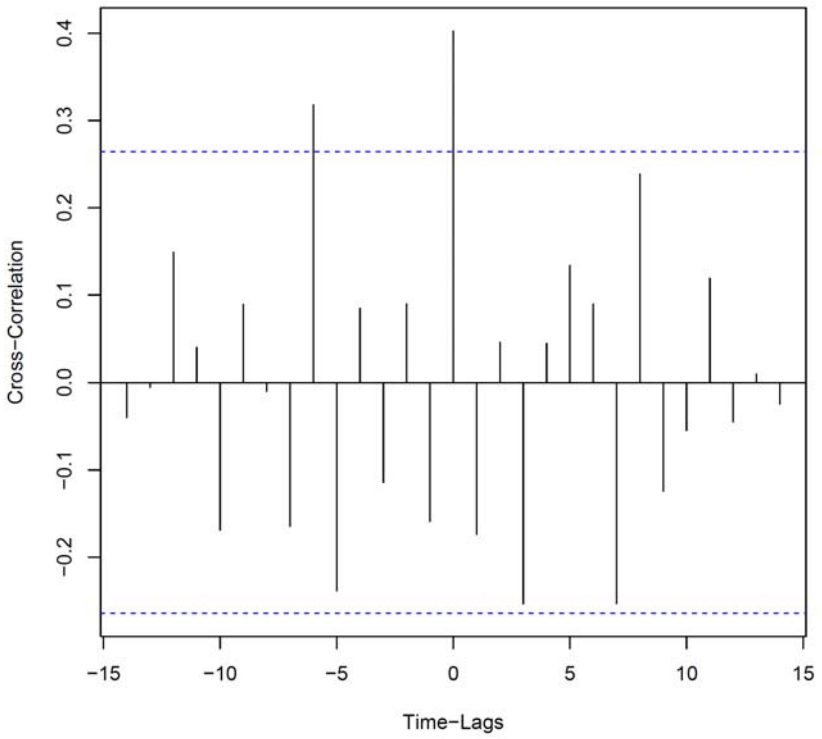

FIGURE 7 Cross-Correlation (CCF) of Machine Learning- and Human-Coded Event Data for Sierra Leone 1999.

FIGURE 8 Spatial Distribution of Machine Learning- and Human-coded Event Data for Sierra Leone 1999.

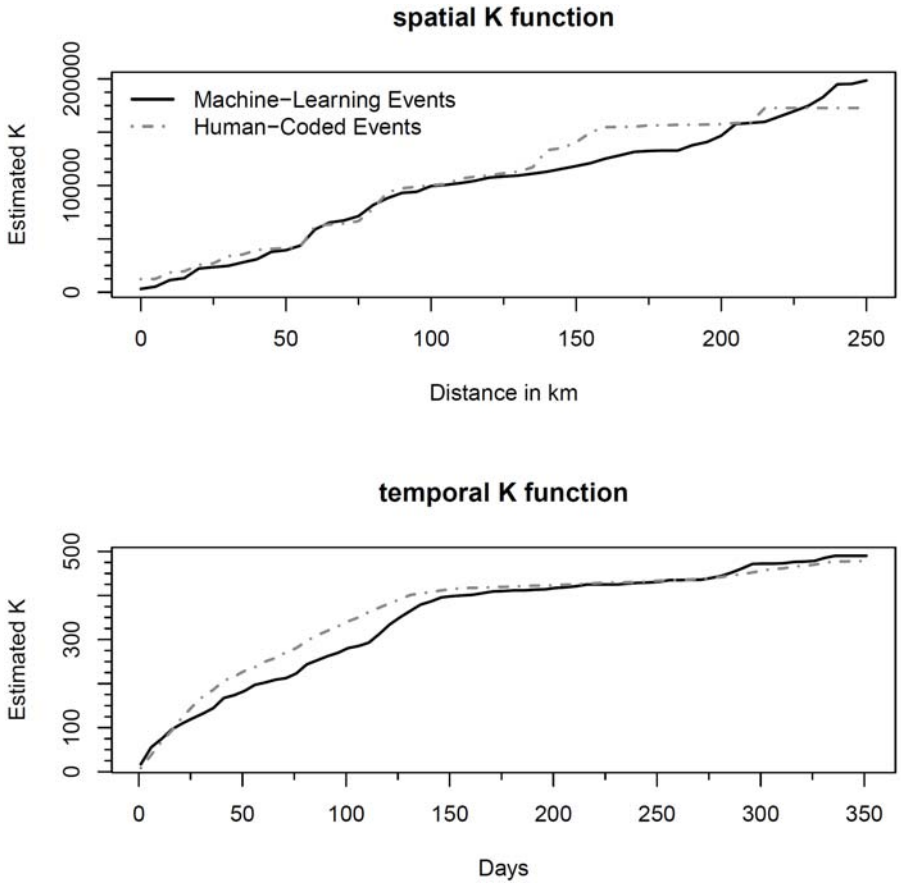

FIGURE 9 Space-Time-K-function Graph of Machine Learning- and Human-Coded EDACS Data for Sierra Leone 1999.

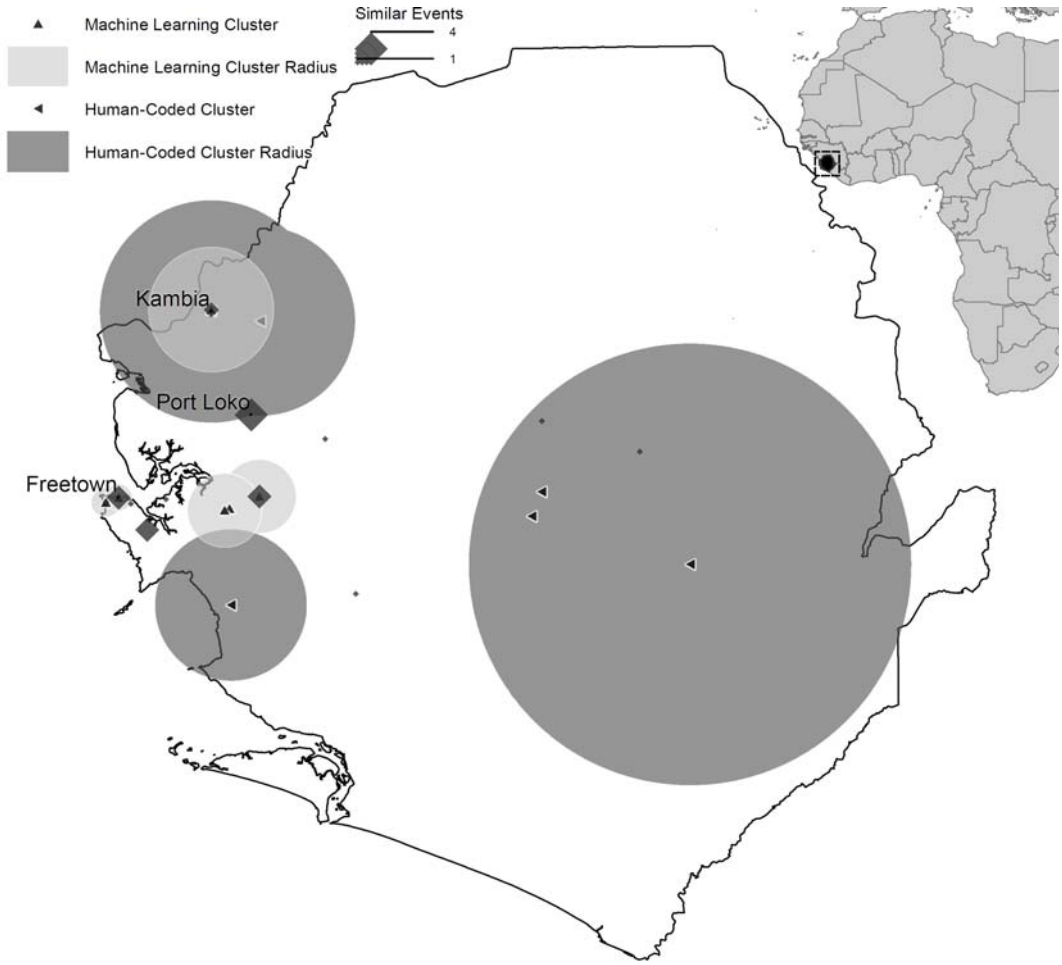

FIGURE 10 SQL-query based similarity matching and Spatiotemporal Permutation SaTScan-statistics of Machine Learning- and Human-Coded Event Data for Sierra Leone 1999.

Endnotes

[1] The project is part of the Collaborate Research Center (SFB) 700 “Governance in Areas of Limited Statehood.”, funded by the German Research Foundation (DFG). We gratefully acknowledge the following people who have been involved in the data gathering process and, therefore, contributed to the success of our project: Michael Spies, Christian Bittner, Katharina Schoenes, Tim Wildemann, Michael Chucholowski.

[2] Mutual violence is defined in EDACS: “as armed interaction between two or more organized groups.” (Codebook (EDACS, 2011: 4)

[3] One-sided violence is defined in EDACS: “as direct unilateral violence by organized groups aimed at civilian or military targets.” (EDACS, 2011: 4).

[4] The toponymic GEOnet Names Server (GNS) database is maintained by the US National Geospatial-Intelligence Agency (NGA) and provides location names and coordinates in the World Geodetic System 1984 (WGS 84) on a global level (NGA, 2011) GNS provides an extensive settlement dataset that is easily accessible at no charge.

[5] EDACS bias, buffering, and overall coding rules and procedures are described in the Codebook < www.conflict-data.org> (online beginning of May 2012).

[6] A more detailed description of the dataset can be found in the download section of our website < www.conflict-data.org> (online beginning of May 2012).

[7] Both decision trees and the meta-classifier AdaBoost are both part of Carnegie Mellon’s MinorThird package (Cohen, 2004), the naive Bayes classifier is from LingPipe’s natural language processing (NLP) libraries (Carpenter, 2010) <http://alias-i.com/lingpipe/>, last accessed 2012/03/22.

[8] We use LingPipe’s commercial implementation due to their high encapsulation and decent documentation, which is free for research use (Carpenter, 2010).

[9] The f-score is most often defined as the harmonic mean of precision – defined as the ratio of relevant items retrieved and all retrieved items – and recall – defined as the ratio of relevant items retrieved and all relevant items (Manning et al., 2008: 156)

[10] Events lasting more than 30 days are supposed to be aggregated, but this proceeding is no guarantee eliminating a bias caused by event-aggregates, rather an arbitrary threshold to minimize possible biases.

[11] The values are smoothed via a locally weighted polynomial-regression (10-procent-window). The lowess-function (Cleveland, 1981) in the R-package {stats} has been applied for that.

[12] The cross-correlation function estimates the degree to which two univariate time.series correlate. For calculation we use the ccf-function (Venables and Ripley, 2002) in the R package {stats}.

[13] A comparative discussion of different space-time-clustering methods is provided by (Norström et al., 2000). They make a case for Kulldorffs Scan-Statistics because other test options like for instance the Knox-test needs space and time thresholds; furthermore Knox-test and the Jacquez-k-nearest-neighbours-test assume that population size does not change over time, whereas Kulldorffs Scan-Statistics can accommodate confounding covariates like population size (Norström, et al., 2000).

[14] Complete Spatiotemporal Randomness (CSTR) implies that there is no stochastic process present in space as well as time (see e.g. Cox and Isham, 1980, Diggle, 2003).

[15] We search for high rates of event clustering, set the parameters for the cluster analysis to 35 percent of the population at risk, and a temporal window of 15 percent of the study period. We further use a circular spatial window shape, aggregate temporally by seven days, and set the temporal cluster size to one day. Finally we run 999 Monte Carlo replications.

[16] Crowd sourcing and crowd seeding are both participative data gathering techniques. Examples for such are: crowd sourcing: International Network of Crisis Mappers (crisismappers.net), crowd seeding Voix des Kivus, latter is conducted by staff of the Columbia University in Eastern Congo: <http://cu-csds.org/wpcontent/uploads/2009/10/Voix-des-Kivus-Leaflet.pdf>.