ORÁCULO

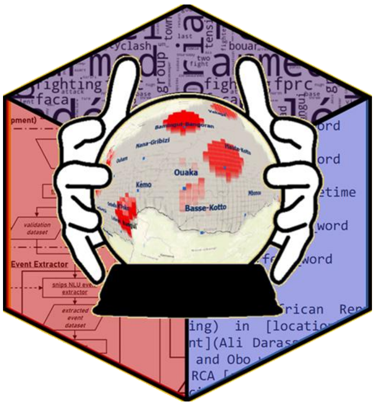
Detection of Spatiotemporal Hot Spots of Conflict-Related Events Extracted from Online News Sources

Artur Jorge Abreu Varanda

Dissertation submitted in partial fulfilment of the requirements for the Degree of Mestre em Ciência e Sistemas de Informação Geográfica (Master in Geographical Information Systems and Science)

NOVA Information Management School

Universidade Nova de Lisboa
NOVA Information Management School



ORÁCULO

Detection of Spatiotemporal Hot Spots of Conflict-Related Events Extracted from Online News Sources

Artur Jorge Abreu Varanda

Dissertation Submitted in partial fulfillment of the requirements for the
Degree of Mestre em Ciência e Sistemas de Informação Geográfica
(Master in Geographical Information Systems and Science)

Advisor: Carina Albuquerque, MSc, NOVA IMS
Supervisor: Victor Lobo, PhD, Escola Naval

21 February 2021

Declaration of Originality

I hereby declare that the work described in this document is my own and not from someone else. All the assistance I have received from other people is duly acknowledged and all the sources (whether published or not) are referenced.

This work has not been previously evaluated or submitted to NOVA Information Management School or elsewhere.

Funchal, 21 February 2021

Artur Jorge Abreu Varanda

[the signed original has been archived by the NOVA IMS services]

Acknowledgments

It is somewhat ironic that the author of a project with the goal of wrestling unstructured text data into geographic information was himself almost wrestled by geography itself: the stretch of Atlantic Ocean between Funchal and Lisbon. However, as the song goes, *“sometimes truth is stranger than fiction”*, and so the development of ORÁCULO coincided with most of my assignment in Madeira. Thus, between the 1 000 kilometers from my loved ones, company command, and an actual, honest-to-god pandemic, one could almost describe the development and writing process as a true *Odyssey* – I know it felt like one!

I did not travel alone. I am also still far away from my destination. But if ORÁCULO made me inch closer to Ithaca, it was due to the significant contributions of a remarkable cast of characters:

First, all colleagues that, amongst the coursework of the Master in Geographic Information Systems and Science, helped me chart a course towards my vision for ORÁCULO: Julian Kuypers, Manuel Demetriades, Marisa Lopes, and Olivier Hofman.

Second, all those that took an interest in ORÁCULO while it was still a simple proof-of-concept, helping me continue the journey with tools, opportunities, or words of encouragement: Colonel Miguel Freire, Professor José Borges, Dr. Camelia Kantor, and Lieutenant-Colonel Nuno Mira.

Then, the ones that “tied me to the mast” of this project in the best possible sense, helping me commit to the path that led to a successful conclusion: my advisor, Professor Carina Albuquerque, my supervisor, Professor Victor Lobo, and my course coordinator, Professor Marco Painho.

Finally, Joana Azinhaes was the ever-present polar star. Though I was oftentimes half an ocean apart, she unfailingly provided me with direction, inspiration, and motivation, propelling me ever forward.

Abstract

Achieving situational awareness in peace operations requires understanding where and when conflict-related activity is most intense. However, the irregular nature of most factions hinders the use of remote sensing, while winning the trust of the host populations to allow the collection of wide-ranging human intelligence is a slow process. Thus, our proposed solution, ORÁCULO, is an information system which detects spatiotemporal hot spots of conflict-related activity by analyzing the patterns of events extracted from online news sources, allowing immediate situational awareness. To do so, it combines a closed-domain supervised event extractor with emerging hot spots analysis of event space-time cubes. The prototype of ORÁCULO was tested on tweets scraped from the Twitter accounts of local and international news sources covering the Central African Republic Civil War, and its test results show that it achieved near state-of-the-art event extraction performance, significant overlap with a reference event dataset, and strong correlation with the hot spots space-time cube generated from the reference event dataset, proving the viability of the proposed solution. Future work will focus on improving the event extraction performance and on testing ORÁCULO in cooperation with peacekeeping organizations.

Keywords: event extraction, natural language understanding, spatiotemporal analysis, peace operations, open-source intelligence.

Resumo

Atingir e manter a consciência situacional em operações de paz requer o conhecimento de quando e onde é que a atividade relacionada com o conflito é mais intensa. Porém, a natureza irregular da maioria das fações dificulta o uso de deteção remota, e ganhar a confiança das populações para permitir a recolha de informações é um processo moroso. Assim, a nossa solução proposta, ORÁCULO, consiste num sistema de informações que deteta "hot spots" espácio-temporais de atividade relacionada com o conflito através da análise dos padrões de eventos extraídos de fontes noticiosas online, (incluindo redes sociais), permitindo consciência situacional imediata. Nesse sentido, a nossa solução combina um extrator de eventos de domínio limitado baseado em aprendizagem supervisionada com a análise de "hot spots" emergentes de cubos espaço-tempo de eventos. O protótipo de ORÁCULO foi testado em tweets recolhidos de fontes noticiosas locais e internacionais que cobrem a Guerra Civil da República Centro-Africana. Os resultados dos seus testes demonstram que foram conseguidos um desempenho de extração de eventos próximo do estado da arte, uma sobreposição significativa com um conjunto de eventos de referência e uma correlação forte com o cubo espaço-tempo de "hot spots" gerado a partir desse conjunto de referência, comprovando a viabilidade da solução proposta. Face aos resultados atingidos, o trabalho futuro focar-se-á em melhorar o desempenho de extração de eventos e em testar o sistema ORÁCULO em cooperação com organizações que conduzam operações paz.

Palavras-chave: extração de eventos, compreensão de linguagem natural, análise espácio-temporal, operações de paz, informações de fontes abertas.

Keywords

Event Extraction

Natural Language Understanding

Spatiotemporal Analysis

Peace Operations

Open-Source Intelligence

Index

List of Figures

List of Tables

List of Acronyms

3R, RRR	Retour, Réclamation et Réhabilitation
ACLED	Armed Conflict Location and Event Data Project
ADM1	Level 1 administrative region
ADM2	Level 2 administrative region
ADM3	Level 3 administrative region
API	Application Programming Interface
AUROC	Area Under the Receiver Operating Characteristics curve
BoW	Bag of Words
CAR	Central African Republic
CINAMIL	Centro de Investigação, Desenvolvimento e Inovação da Academia Militar
CRF	Conditional Random Fields
CRISP-DM	Cross Industry Standard Process for Data Mining
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DSTTM	Dynamic Spatio-Temporal Tweet Mining
EPSG	European Petroleum Survey Group
ETC	Emergency Telecommunications Cluster
FACA	Forces armées centrafricaines
FDPC	Front Démocratique du Peuple Centrafricain
FOMUC	Force multinationale en Centrafrique
FPRC	Front Populaire pour la Renaissance de la Centrafrique
GCNN	Graph Convolutional Neural Networks
GEOINT	Geospatial Intelligence
GIS	Geographic Information Systems
HDI	Human Development Index
HUMINT	Human Intelligence
ICT	information and communications technology
1IDP	Internally Displaced Persons
JMAC	Joint Mission Analysis Center
JOC	Joint Operations Center
LDA	Latent Dirichlet Allocation
LR/MLN	Linear Regression with Maximum Likelihood Estimation model/Markov Logic Networks
LRA	Lord’s Resistance Army
MAgP	Mean Average Generalized Precision
MCJ	Mouvement des Libérateurs Centrafricains pour la Justice
MICOPAX	Mission de consolidation de la paix en Centrafrique

MINUSCA	Mission multidimensionnelle intégrée des Nations unies pour la stabilisation en Centrafrique
MISCA	Mission Internationale de Soutien à la Centrafrique sous Conduite Africaine
MLN	Markov Logic Networks
MLT	Mission Headquarters and Leadership Team
MPC	Mouvement Patriotique pour la Centrafrique
NATO	North Atlantic Treaty Organization
NLP	Natural Language Processing
NLU	Natural Language Understanding
NP	Noun Phrase
OCHA	Office for the Coordination of Humanitarian Affairs
OPTICS	Ordering Points To Identify the Clustering Structure
OSINT	Open-Source Intelligence
PCS	Projected Coordinate System
POS	Parts-of-Speech
R&D	Research and Development
RJ	Révolution et Justice
RJDH	Réseau des Journalistes pour les Droits de l'Homme en Centrafrique
RPRC	Rassemblement Patriotique pour le Renouveau de la Centrafrique
SAGE	Situational Awareness Geospatial Enterprise
SIGINT	Signals Intelligence
SOM	Self-Organizing Maps
SRSG	Special Representative of the Secretary General of the UN
SVM	Support Vector Machines
TF-IDF	Term Frequency–Inverse Document Frequency
UN	United Nations
UNAMID	United Nations African Union Mission in Darfur
UPC	Union pour la Paix en Centrafrique
USGIF	United States Geospatial Intelligence Foundation
UTM	Universal Transverse Mercator
VP	Verb Phrase
WGS	World Geodetic System

1 Introduction

1.1 Problem Definition: Situational Awareness in Peace Operations

No captain can do very wrong if he places his ship alongside that of the enemy.
Horatio Nelson, posthumous victor of the Battle of Trafalgar (1805)

Whether an organism or an organization, being aware of one's environment is a condition for effective decision-making, to the point where impairments in sensing capabilities are generally regarded as severe disabilities. This holds true for military decision-making and has led to the development of military intelligence as a process, as a product and as a set of dedicated people and resources (ATP-321, 2009). Though well-developed, most of those processes and organizations evolved during the 20th Century to answer intelligence requirements about similarly organized adversaries (symmetric warfare). However, current patterns of conflict are dominated by vastly different settings, namely civil (intrastate) wars (Strand, Rustad, Urdal, & Nygård, 2019) in which at least one of the factions operates hidden amongst the civilian population[1] (asymmetric warfare). This has changed *situational awareness* – defined as "the knowledge of the battlespace necessary to make well-informed decisions" (AAP-6, 2015) – from a mostly physical challenge of finding the enemy to a mostly human challenge of finding where and when the enemy usually operates – a challenge which approaches that faced daily by law enforcement agencies. For instance, the best remote sensing can detect armed personnel hiding in forested, rugged terrain, but cannot distinguish terrorists from bystanders in a small village before the start of a terrorist attack, nor even warn that the village is a frequent stage of enemy activity. So, drawing from the quote above, *how can commanders *not* do very wrong if they cannot even find the seas through which the enemy navigates?*

Like in law enforcement, the solution lies in accessing the information possessed by the "sea", meaning "the populations themselves": if rebels hide amongst the people, the people are the best way of finding where the rebels operate[2]. Accessing that information, however, requires trust, which is more easily won (and less easily lost) by

[1] The wars in Vietnam (1955-1975), Afghanistan (2001-ongoing), the Portuguese Colonial War (1961-1974) and the settings of most United Nations (UN)-led peace operations, such as the Central African Republic Civil War (2012-ongoing), are examples of such "small wars".

[2] This is the corollary of the famous prescription of Mao Zedong: "The guerrilla must move amongst the people as a fish swims in the sea" (paraphrased from *On Guerrilla Warfare*, 1965).

national forces fighting irregulars in their own territory than by foreign peacekeepers (Berman, Felter, & Shapiro, 2018) trying to impose a safe and stable environment on a chaotic conflict with multiple factions. Fortunately, when news organizations continue to operate unhindered throughout the conflict, conflict-related events will be reported, providing an important contribution to the situational awareness of the peacekeeping force. This practice falls within the widely adopted concept of OSINT (Open-Source Intelligence) (ATP 2-22.9, 2012).

Nevertheless, the use of open sources compounds the situational awareness challenge: better and cheaper platforms (smartphones) and infrastructure (4G/5G networks, social media platforms) have increased the amount of news sources and information[3] to the point where available data greatly exceeds the analysis capacity of unassisted humans – especially in high-pressure environments where available manpower is low. However, having more data than one can analyze is not a challenge exclusive to military intelligence, and data mining solutions can be adapted to help solve this problem – the **research problem**, which we can state as:

How to detect areas and time periods of significant conflict-related activity in peace operations using open-source information, and with minimal human intervention?

1.2 Project Goal and Relevance: ORÁCULO

To tackle the research problem, we propose to combine two emerging technologies: Emerging Hot Spots Analysis (ESRI, 2020a) and Event Extraction (Sundheim & Chinchor, 1993). Event Extraction refers to the use of natural language

[3] Statistics compiled by the International Telecommunication Union (2020) show how the worldwide percentage of internet users has climbed steadily, from 8.0% in 2001 to an estimated 53.6% in 2019, increasing linearly by about 2.5% of the global population each year – both in developed and developing countries. In comparison, the amount of data produced by these users is growing exponentially: the total size of the internet in terms of data has grown from 4.4 zetabytes (1×10^{21} bytes) in 2013 to little over 20 zetabytes in 2017 and over 44 zetabytes in 2020, with one report commissioned by the data storage company Seagate predicting that the global size will reach 175 zetabytes in 2025 (Reinsel, Gantz, & Rydning, 2018). Though the average percentage of internet users in Africa in 2019 was still an estimated 28.2% (and just 4.3% for the Central African Republic), as that percentage climbs towards 100%, the sheer size of Africa's population means that that small percentage will produce a disproportionate amount of total data.

Likewise, as the internet continues to increase in size and importance, so do online news sources. Research by the Pew Research Center (2019) conducted in the United States illustrates that trend by highlighting a steady linear increase in the numbers of newsroom employees in the digital-native sector (online news sources). When considering this information in tandem with the tendencies outlined above, it is therefore reasonable to expect that the importance of online news outlets as sources of open-source information will only increase in the future – even in developing countries.

understanding engines to detect events in unstructured text data and to retrieve their arguments (*e.g.*, Li, Cheng, He, Wang, & Jin, 2019). For example, the event type (*action*), its *location* and *time*, the associated *agent* and *target*, and the event's immediate *effects*. Emerging Hot Spots Analysis describes the combined use of Geographic Information Systems (GIS), spatial statistics and time series analysis to detect spatial and temporal hot spots of events and their temporal trends. In short, we propose to adapt the proven concept of identifying hot spots of criminal activity ("hot spots policing") in a city (*e.g.*, Telep & Weisburd, 2016) to the problem of identifying hot spots of conflict-related activity in a country, using events extracted from online open news sources to feed the analysis.

In this project, we propose to implement the proposed solution in a prototype, "ORÁCULO", and to test it on the Central African Republic (CAR) Civil War (2012-ongoing), the setting of the *Mission Multidimensionnelle Intégrée des Nations Unies pour la stabilisation en Centrafrique* (UN Multidimensional Integrated Mission for the Stabilization of the Central African Republic – MINUSCA). We chose this use case because we consider it representative of current and future peace operations, but also because we are motivated by the benefits which the outcome of this project could bring to the peacekeepers – which comprise a significant number of Portuguese Army and Air Force troops[4] – and to the Central African civilians. To conclude, we can define the **project goal** as:

Developing and testing the prototype of a system, ORÁCULO, capable of detecting spatiotemporal hot spots of conflict-related events extracted from online news sources.

If proven successful, the resulting system can be adapted to other conflicts, providing immediate situational awareness to peacekeeping organizations while relations with the host nations are not well-developed enough to allow trustworthy intelligence collection from the populations themselves. Furthermore, expected advances in information technology and infrastructure and the associated increase in available data should increase the accuracy and quality of the proposed system's output even further, cementing the relevance of this approach and of this project.

[4] This research is part of an overarching research and development (R&D) project, project ORÁCULO, whose end goal is to increase the situational awareness of Portuguese forces involved in peace operations. The project is funded by the Portuguese Army through the Portuguese Military Academy Research and Development Center (CINAMIL) (Academia Militar, 2020). At the time of writing, besides several national presentations, project ORÁCULO was presented at two United States Geospatial Intelligence Foundation (USGIF) events: GEOINTegration Summit 2019 and GEOConnect Series 2020 (link).

1.3 Project Structure and Research Tasks: CRISP-DM

Since the research problem can be classified as a data mining problem[5] (albeit a geospatial one[6]), the Cross Industry Standard Process for Data Mining (CRISP-DM) (Wirth & Hipp, 2000) was selected as the research process model, *i.e.*, the process responsible for guiding the research efforts towards the research goal. CRISP-DM is often described as "by far the most widely-used analytics process standard (…), flexible enough to suit many analytic styles" (Brown, 2015). Its flexibility allows the process to adapt well to geospatial data mining problems (Pretorius & Matthee, 2006), which, in combination with it being the industry standard, led to its adoption as the research process model. It is a data-centric, four-level cycle comprised of six phases (first level) (Figure 1, page 5), with each phase consisting of several generic tasks (second level) (Chapman, et al., 2000). The phases and generic tasks are common to all data mining projects, so specific specialized tasks (third level) and processes (fourth level) must be defined for each project (Figure 2, page 5). In the following pages, Table 1 (pages 6 to 8) describes the mapping between the chapters and sections of this report, CRISP-DM phases and generic tasks, and research (specialized) tasks.

As for the report itself, it begins with Chapter 2, "Context Understanding", which briefly describes the project context: MINUSCA and their efforts to maintain situational awareness. Then, Chapter 3, "Literature Review and State-of-the-Art" details the state-of-the-art of spatiotemporal analysis and event extraction and describes existing work on the research problem, allowing the selection of technologies and methods for ORÁCULO. The implementation of these methods and technologies in the project is explained in Chapter 4, "System Architecture and Implementation", which describes the project methodology and data and the development and structure of ORÁCULO and its components. Finally, Chapter 5, "Results and Discussion", explains test procedures,

[5] Meaning, a problem solvable through the application of data mining. In turn, data mining – also known as Knowledge Discovery in Data/Databases – can be defined as "nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data" (*e.g.*, Fayyad, Piatetsky-Shapiro, & Smyth, 1996). ORACLE Corporation – an industry leader in database technology – describes its key properties as "the automatic discovery of patterns", "the prediction of likely outcomes", "the creation of actionable information", and a "focus on large data sets and databases" (ORACLE Corporation, 2013).

[6] "Geospatial data mining is a subfield of data mining concerned with the discovery of patterns in geospatial databases" (*e.g.*, National Research Council, 2003). Geospatial/geographic databases (abbreviated to "geodatabases") are databases capable of storing and querying geographic data (*e.g.*, Longley, Goodchild, Maguire, & Rhind, 2005). "Spotting spatial-temporal clusters in crime data" is described by Yuan, Buttenfield, Gahegan, & Miller (2004) as a typical problem of geospatial data mining.

showcases test results, and discusses them considering the project context and state-of-the-art. The report concludes with Chapter 6, "Conclusion", which assesses whether the project goal was achieved, identifies its limitations, and proposes future work on the research problem.

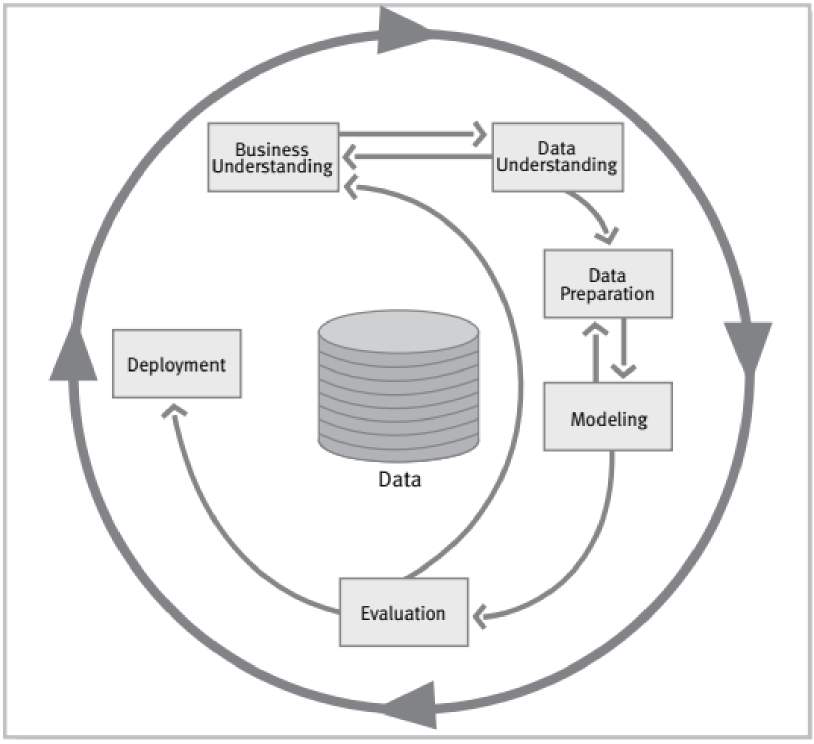


Figure 1. Outline of the CRISP-DM. Note how it is a cyclical process, a property particularly important to the research problem. *Source:* (Chapman, et al., 2000).

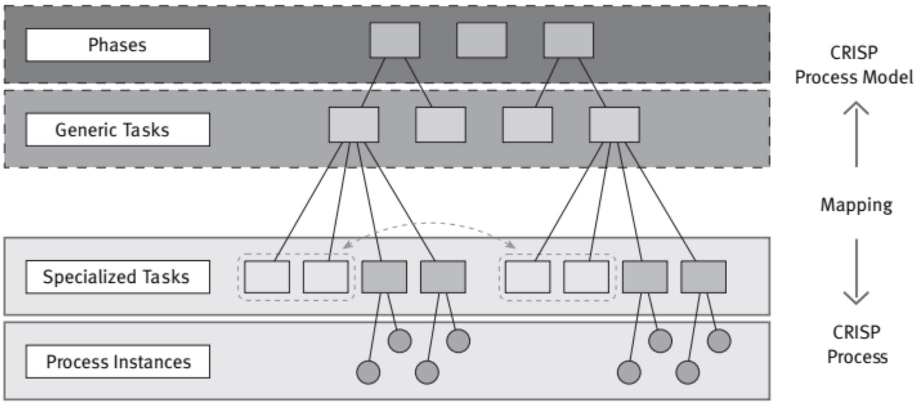


Figure 2. Four-level breakdown of the CRISP-DM methodology. *Source:* (Chapman, et al., 2000).

Table 1. Mapping of Report Chapters and Project Research Tasks to CRISP-DM phases and generic tasks. Project Research Tasks are Specialized Tasks in the context of the CRISP-DM process model. Also note that phases and tasks prescribed by CRISP-DM represent an ideal sequence of events, but changes in task order are allowed if the context dictates: for instance, the generic task of "clean data" was placed before "select data" to reflect the architecture of the ORÁCULO system.

Report Chapter	Report Section	CRISP-DM Phase	CRISP-DM Generic Task	Project Research Tasks
Chapter 2. Context Understanding	2.2. Situational Awareness of MINUSCA	Business Understanding	Determine Business Objectives	Describe how MINUSCA maintains situational awareness.
			Assess Situation	Define the limitations of the MINUSCA situational awareness process.
Chapter 3. Literature Review and State-of-the-Art				Describe the technological state-of-the-art and the possible solutions to the research problem.
Chapter 4. System Architecture and Implementation	4.1. System Architecture		Determine Data Mining Goals	Define the requirements of the ORÁCULO system.
			Produce Project Plan	Define *conflict-related event* and its arguments in the context of ORÁCULO. Describe the components of the ORÁCULO system.
	4.1. Data and Preprocessing	Data Understanding	Collect Initial Data	Obtain *near-ground truth* open-source event dataset (ACLED) covering the CAR Civil War during the scraping period (2019-2020). Scrape online news sources (Twitter accounts) covering the CAR Civil War during the scraping period (2019-2020).
			Describe Data	Describe the attributes of the ACLED event dataset. Annotate the scraped tweet dataset to create the training/validation dataset.
			Explore Data	Perform exploratory data analysis (content, temporal data, spatial data) on the ACLED event dataset and training/validation dataset.
			Verify Data Quality	Ensure thematic, temporal, and spatial consistency between the training/validation dataset and the ACLED event dataset.

Report Chapter	Report Section	CRISP-DM Phase	CRISP-DM Generic Task	Project Research Tasks
Chapter 4. System Architecture and Implementation	4.3 Event Extractor 4.4. Event Dataset Supervision	Data Preparation	Clean Data	Preprocess and translate scraped tweet dataset.
			Select Data	Detect conflict-related events in the scraped tweet dataset.
			Construct Data	Extract event arguments from the detected events. Geocode extracted event locations.
			Integrate Data	Merge redundant detected and extracted events into unique events.
	4.5 Hot Spots Detector		Format Data	Generate the event geodatabase from the merged events dataset.
		Modelling	Select Modelling Technique	Define parameters of the emerging hot spots analysis performed by ORÁCULO.
			Build Model	Perform emerging hot spots analysis on the event space-time cubes.
Chapter 5. Results and Discussion	5.1 Results		Generate Test Design	Define test procedures, performance metrics, and validation and ground truth datasets.
			Assess Model	Measure the performance of ORÁCULO by testing its components and functions using the validation and ground truth datasets.

Report Chapter	Report Section	CRISP-DM Phase	CRISP-DM Generic Task	Project Research Tasks
Chapter 5. Results and Discussion	5.2 Discussion	Evaluation	Evaluate Results	Compare results with the state-of-the-art and with the project context.
Chapter 6. Conclusion and Recommendations	6.1 Assessment of Project Goal		Review Process	**Discuss achievement of the project goal.**
	6.2 Limitations			Comment on the research constraints.
	6.3 Future Work		Determine Next Steps.	Propose further research on the research problem.
		Deployment	Plan Deployment	Propose deployment plan for the ORÁCULO system.
			Plan Monitoring and Maintenance	Propose monitoring and maintenance plan for the ORÁCULO system.
			Final Report	*(The present document is the final report.)*
			Review Project	**Discuss the state of the project.**

2 Context Understanding

CHAPTER GOALS

The goals of Chapter 2 are explaining the context and objectives of MINUSCA, the processes it uses to maintain situational awareness, and their current limitations. It maps to the "Determine Business objectives" and "Assess Situation" tasks of the "Business Understanding" phase of CRISP-DM.

2.1 From the CAR Civil War to MINUSCA

2.1.1 CAR Civil War

Having been the stage of 12 international peacekeeping and peacebuilding interventions since 1997 (Olin, 2015), the CAR provides a good setup to test the proposed solution to the research problem. The latest conflict, dubbed the "CAR Civil War", began in December 2012 with the creation of the "Séléka", a loose alliance of rebel factions from the Northeastern CAR, fighters from the Darfur, and Chadian support, united mostly by their common rejection of the regime of then-President François Bozizé. Shortly after its inception, the Séléka took over the sparsely populated and almost ungoverned northeastern *préfectures* (level 1 administrative regions) and started an offensive towards the capital, Bangui, which was stopped at the last moment due to the intervention of the peacekeepers of the MICOPAX[7] (*Mission de consolidation de la paix en Centrafrique* – Mission for the Consolidation of Peace in the Central African Republic). Recognizing the danger posed by the Séléka, Bozizé launched the Libreville peace talks in early 2013 and agreed on a coalition government with the rebel factions, but the collation soon crumbled. Finally, on 22 March 2013, the Séléka launched a new attack on Bangui, but this time the MICOPAX peacekeepers did not intervene. Forty-eight hours later, Bozizé had fled Bangui and the rebel alliance had taken over the government of the CAR, with rebel leader Michael Djotodia proclaiming himself President (Smith S. W., 2015).

[7] In 2008, the Economic Community of Central African States took over the Multinational Force in the Central African Republic (FOMUC, created in 2002 and sponsored by the Economic and Monetary Community of Central African States) and renamed it MICOPAX. It was made up of 2 300 soldiers and 380 policemen from neighboring African countries, including Chad (Olin, 2015).

However, rather than putting an end to the fighting, the takeover by the Séléka resulted in its escalation. Most factions in the Séléka came from the mostly Muslim Northeast, and while their cause was political – they opposed the Bozizé regime –, their takeover was perceived as a Muslim takeover of the CAR by the mostly Christian Central African population[8]. In turn, this drove the self-defense militias from the mainly Christian *préfectures* to evolve into a loosely coordinated rebel faction known as the "Anti-Balaka", declaring the intention of deposing the Muslim-dominated government. Despite the dissolution of the Séléka in September 2013, the resignation of Djotodia in December 2013, the creation of an interim government approved by the Anti-Balaka, and the concurrent beginning of the French-led Operation Sangaris and the United Nations (UN) *Mission Internationale de Soutien à la Centrafrique sous Conduite Africaine* (MISCA), a fully-fledged civil war between the Séléka successor factions, Anti-Balaka groups, local rebel factions, and the Central African government was well under way (see Table 2, page 11, for a rundown of the warring factions). Accordingly, by March 2014 it was estimated that sectarian violence had led to some 430 000 internally displaced persons (IDP) and to some 419 000 refugees, and that 2.5 million people – out of a population of 5.8 million – needed urgent humanitarian help (OCHA, 2015), leading the Human Development Report 2015 (created from 2014 data) to classify the CAR as the second-last country in the world in terms of Human Development Index (HDI) (United Nations Development Programme, 2015).

2.1.2 MINUSCA

In response to the increasing violence, the UN chose not to renew the mandate of MISCA and established the MINUSCA in April 2014, a multinational, multidimensional peace operation with the mandate (goals) of protecting civilians, ending the sectarian violence, disarming, demobilizing, and reintegrating the rebel factions, and fostering national governance (United Nations Security Council, 2014). By March 2020, MINUSCA comprised 13 252 personnel, with 19 countries contributing more than 100 personnel. Progress has been mixed: a UN-mediated Political Agreement for Peace and Reconciliation ("Khartoum Agreement")[9] was signed on 6 February 2019 by the CAR government and by 12 different rebel groups (United Nations Security Council, 2019a)

[8] A former French protectorate granted independence in 1960, the CAR inherited the colonial borders, which "glue" together a Christian majority and a significant Muslim minority (8.5% according to 2010 estimates) based on the northeastern *préfectures* (Central Intelligence Agency, 2020).

[9] A peace agreement had already been attempted at the Bangui National Forum (15 May 2015), but it did not result in any lasting peace.

(Table 2, page 11), but violations of the Agreement by the signatory parties have prompted the UN to renew the mandate of MINUSCA for another year (United Nations Security Council, 2019b), and though the latest Report of the Secretary-General on the Central African Republic (United Nations Security Council, 2020) reinforces the perception of slow progress, it also explicitly warns about the fragility of the situation. This assessment is echoed by the third-party report by Lise Howard (2019) on behalf of the International Peace Institute, which describes the conclusion of MINUSCA as "not in sight" essentially because the rebel groups continue to operate due to flaws in the application of military power, spoiling political and humanitarian progress. It is therefore reasonable to expect new mandate extensions for the foreseeable future.

Table 2. Signatory factions of the Khartoum Agreement (United Nations Security Council, 2019a) and other important rebel groups currently operating in the CAR (Dukhan, 2017).

Faction		Notes
MINUSCA		
CAR Government		Currently headed by Faustin-Archange Touadera, an independent candidate.
Rebel groups that signed the Khartoum Agreement	*Front Démocratique du Peuple Centrafricain* (**FDPC**)	Séléka successor faction.
	Front Populaire pour la Renaissance de la Centrafrique (**FPRC**)	Séléka successor faction.
	Mouvement Patriotique pour la Centrafrique (**MPC**)	Séléka successor faction.
	Union pour la Paix en Centrafrique (**UPC**)	Séléka successor faction.
	Séléka Rénovée	Séléka successor faction.
	Rassemblement Patriotique pour le Renouveau de la Centrafrique (**RPRC**)	Séléka successor faction.
	Mouvement des Libérateurs Centrafricains pour la Justice (**MLCJ**)	Based on the Northeastern *préfectures* and linked to the Kara ethnic group.
	Retour, Réclamation et Réhabilitation (**RRR** or **3R**)	Based on the Northwestern *préfectures* and linked to the Fulani ethnic group.
	Anti-Balaka – Mokom Branch	
	Anti-Balaka – Ngaïssona Branch	
	Révolution et Justice – Belanga Branch (**RJ-Belanga**)	Based on the Northwestern *préfectures*.
	Révolution et Justice – Sayo Branch (**RJ-Sayo**)	Based on the Northwestern *préfectures*.
Other important rebel groups that operate in the CAR	Lord's Resistance Army (**LRA**)	Based on the Southeastern *préfectures*.

In terms of organization, MINUSCA generally adheres to the template for multidimensional peace operations set by UN doctrine (MINUSCA, 2020) (UN Department of Peacekeeping Operations, 2008), comprising a *ca.* 10 000-strong military component, a ca. 2 000-strong police component and a civilian component of ca. 1 000 experts. The components and their respective capabilities operate together in the three geographic sectors (West, Center and East, loosely divided along *préfecture* boundaries) (Figure 3, page 14), and their action is coordinated by the Mission Headquarters and Leadership Team (MLT) according to the guidance of the Special Representative of the Secretary General of the UN (SRSG), which holds overall responsibility for the UN efforts in the CAR. The MLT further comprises (amongst other elements) two groups which directly support the integration of component efforts towards the mandate goals: the Joint Operations Center (JOC) and the Joint Mission Analysis Center (JMAC).

2.2 Situational Awareness of MINUSCA

2.2.1 Processes

The JMAC and the JOC are the top groups responsible for situational awareness and intelligence (UN Department of Peacekeeping Operations, 2008): the JOC "collates situation reports and operational information from all mission sources to provide current situational awareness for the mission", while the JMAC "provides integrated analysis of all sources of information to assess medium- and long-term threats to the mandate and to support MLT decision-making". Thus, neither the JOC nor the JMAC are responsible for *collecting* intelligence. Rather, JMAC *directs* the collection efforts of the components, both *process* the result – the JOC into a common operational picture, the JMAC into detailed reports and forecasting –, and both *disseminate* it to the MLT and to the components (Military Peacekeeping-Intelligence Handbook, 2019). As for the process of intelligence collection itself, peace operations rely mostly on intelligence collected from human sources (HUMINT), *e.g.*, patrol reports and intelligence provided by local informers (United Nations, 2019).

To illustrate the MINUSCA intelligence cycle, picture the following scenario (Figure 4, page 15): during a routine patrol in the vicinity of Bouar, soldiers of the Bangladeshi battalion are tipped by a farmer about a punitive action which the RRR rebel group conducted last night against the local population, resulting in three wounded civilians. The patrol leader includes this information on the patrol report, which is then forwarded to the Bangladeshi battalion staff, sector west staff, and force headquarters staff through their respective intelligence sections (S2, G2, U2), eventually reaching the JOC. Then, the JOC can input the reported event on the overall database, which is fused by the JMAC with intelligence collected from other sources (*e.g.*, OSINT, remote sensing) to analyze patterns and clusters of events, identifying trends in the overall situation. The result is then reported to the MLT.

One recent innovation is the adoption of the Situational Awareness Geospatial Enterprise (SAGE) information management system (Figure 5, page 16). SAGE is a GIS-based event reporting platform "reverse-engineered" from the multitude of improvised solutions developed independently in several peace operations (Expert Panel on Technology and Innovation in UN Peacekeeping, 2015). It is based on the Ushahidi platform (see 3.1) and aims to shorten the time between the event collection and integration into the overall, mission-level database by allowing event reports to be sent by text message, email, Twitter, or web forms directly to the JOC. (In the scenario of the Bangladeshi patrol, the patrol leader could send a text message to the JOC containing the reported event as soon as the tip was collected from the population). Analysts can then process the reports in the database into conflict-related events, which are displayed in a map of the CAR using web GIS. The resulting visualization is then made accessible to every relevant stakeholder (*e.g.*, JMAC, MLT, headquarters of police and military components at the various levels).

SAGE is still in its infancy, having been introduced in MINUSCA in 2018. Despite an early "lack of buy-in from some military and police components" in the missions where it was first tested (Expert Panel on Technology and Innovation in UN Peacekeeping, 2015), the project has progressed well enough for the 2019-2020 MINUSCA budget to provide funds for eight extra Joint Operational Officers, one for each new subnational-level JOC[10]. Their responsibilities are the "implementation of the SAGE database and mission common operational picture projects" (United Nations General Assembly, 2019), including providing the training on reporting through the SAGE platform – which is essential to make the most of its capabilities. No known plans exist to expand SAGE to automatically integrate imagery and open-source information.

[10] The locations of the new JOC are: Bambari, Bangassou, Berberati, Birao, Bossangoa, Ndélé, Obo, and Paoua.

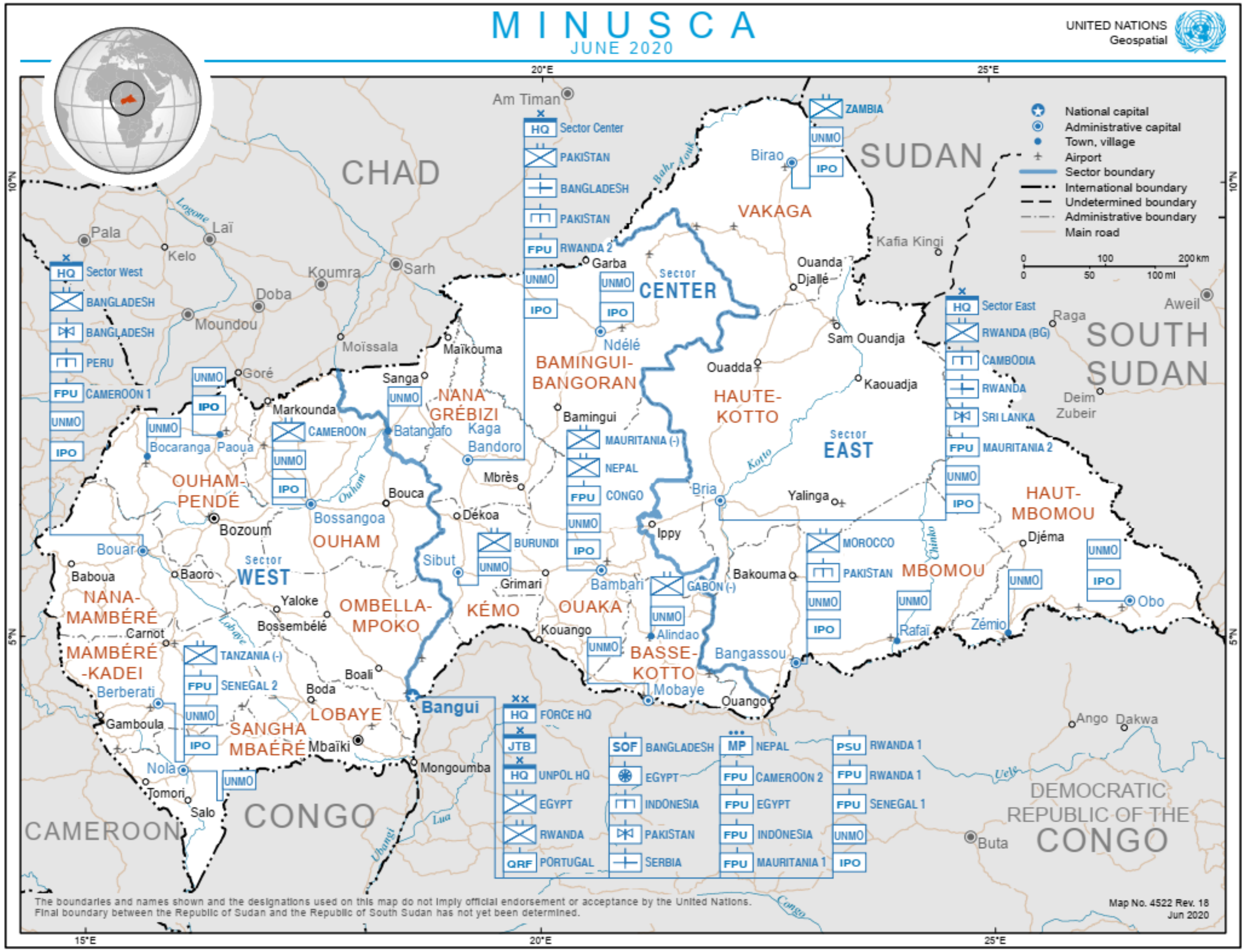


Figure 3. Map of the spatial distribution of MINUSCA forces in June 2020. *Source:* (United Nations Geospatial Information Section, 2020).

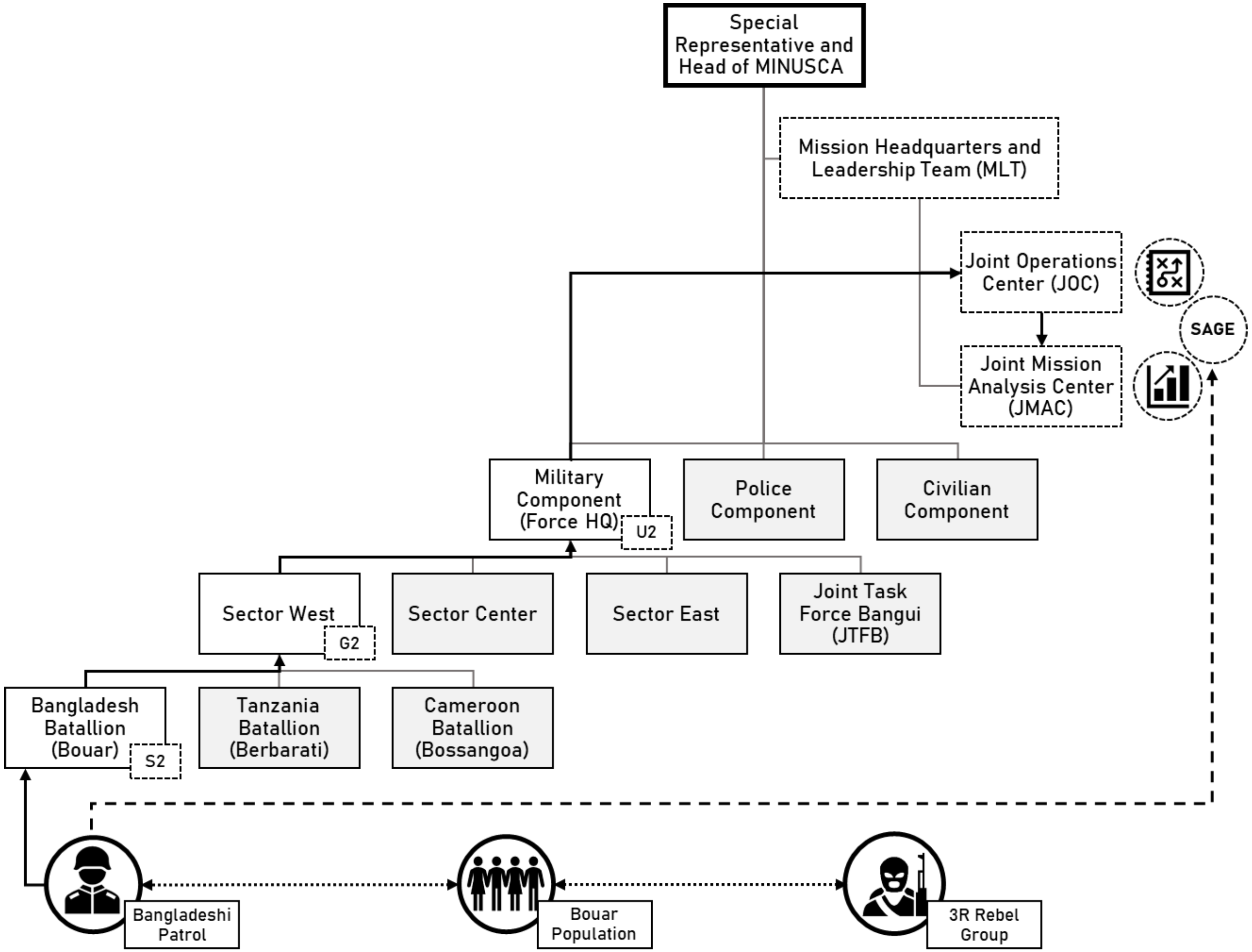
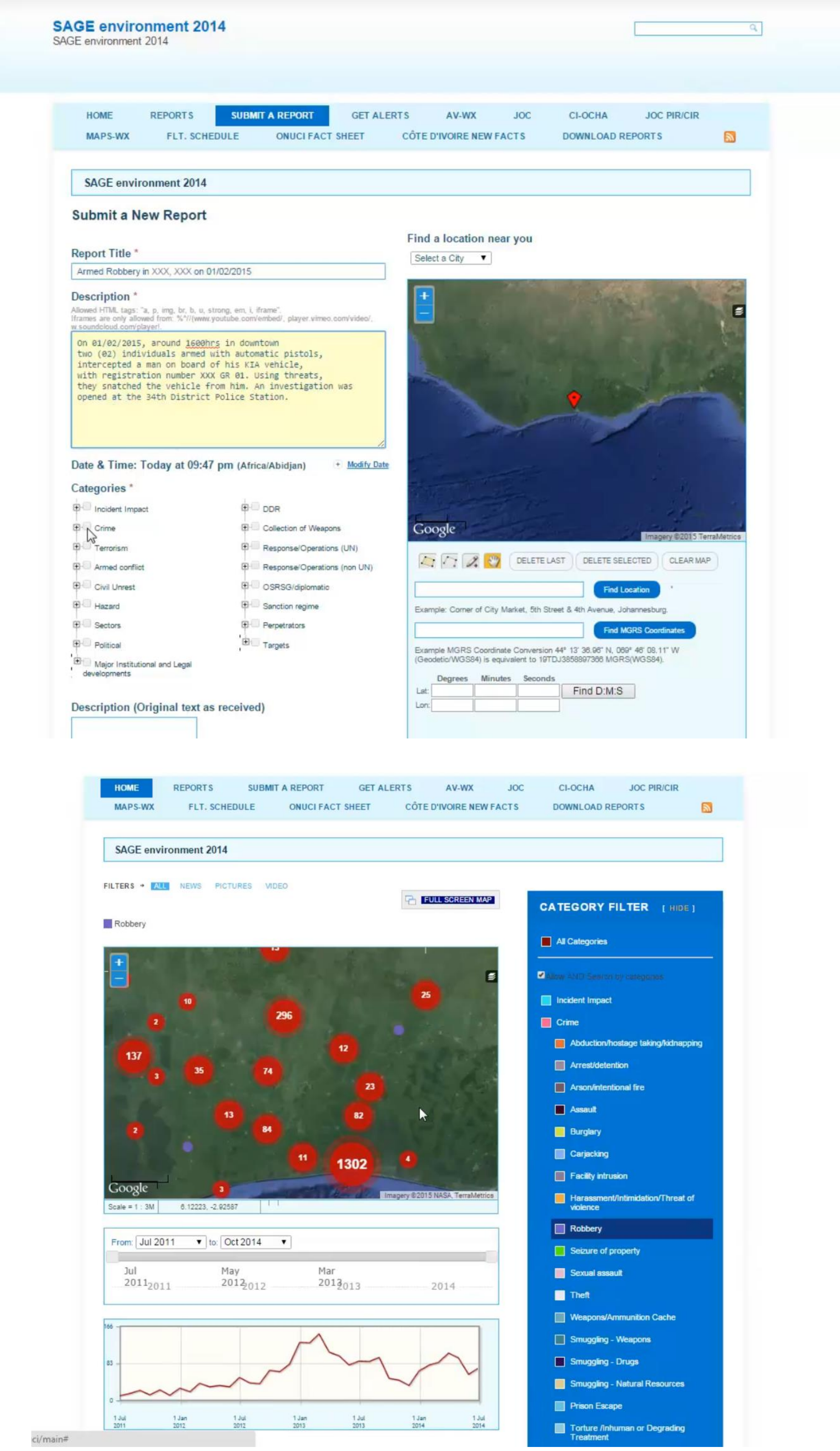

Special Representative and Head of MINUSCA
Mission Headquarters and Leadership Team (MLT)
Joint Operations Center (JOC)
SAGE
Joint Mission Analysis Center (JMAC)
Military Component (Force HQ)
U2
Police Component
Civilian Component
Sector West
G2
Sector Center
Sector East
Joint Task Force Bangui (JTFB)
Bangladesh Batallion (Bouar)
S2
Tanzania Batallion (Berbarati)
Cameroon Batallion (Bossangoa)
Bangladeshi Patrol
Bouar Population
3R Rebel Group

Figure 4. Illustration of the scenario where a Bangladeshi patrol succeeds in acquiring information about a punitive action against the population conducted by a local branch of the RRR rebel group. The SAGE system and the advantages it brings are described below. *Organization chart adapted from:* (UN Department of Peacekeeping Operations, 2008)



Report Title *

Armed Robbery in XXX, XXX on 01/02/2015

Description *

Allowed HTML tags: "a, p, img, br, b, u, strong, em, i, iframe".
Iframes are only allowed from: %^//(www.youtube.com/embed/, player.vimeo.com/video/, w.soundcloud.com/player/!.

```
On 01/02/2015, around 1600hrs in downtown
two (02) individuals armed with automatic pistols,
intercepted a man on board of his KIA vehicle,
with registration number XXX GR 01. Using threats,
they snatched the vehicle from him. An investigation was
opened at the 34th District Police Station.
```

Date & Time: Today at 09:47 pm (Africa/Abidjan) Modify Date

Categories *

Description (Original text as received)

Example: Corner of City Market, 5th Street & 4th Avenue, Johannesburg.

Example MGRS Coordinate Conversion 44° 13' 36.96" N, 069° 46' 08.11" W (Geodetic/WGS84) is equivalent to 19TDJ3858897366 MGRS(WGS84).

	Degrees	Minutes	Seconds
Lat:			
Lon:			

Figure 5. Screenshots of the SAGE information management system. Above, the form used to submit an event report; below, the resulting web GIS visualization. *Source:* (Manning, 2018).

2.2.2 Limitations

Despite the introduction of SAGE, the situational awareness of MINUSCA is hindered by two structural limitations related to current intelligence collection processes:

- First, local tips depend on the local disposition and trust towards MINUSCA, which polls have shown to be unfavorable and significantly inferior to the disposition and trust towards local police, gendarmerie and armed forces (Vinck, Pham, Balthazard, & Magbe, 2019) (Figure 6);
- Second, it lacks the manpower and the resources to permanently patrol every city and town, limiting the events collected by patrol reports to the areas being patrolled[11].

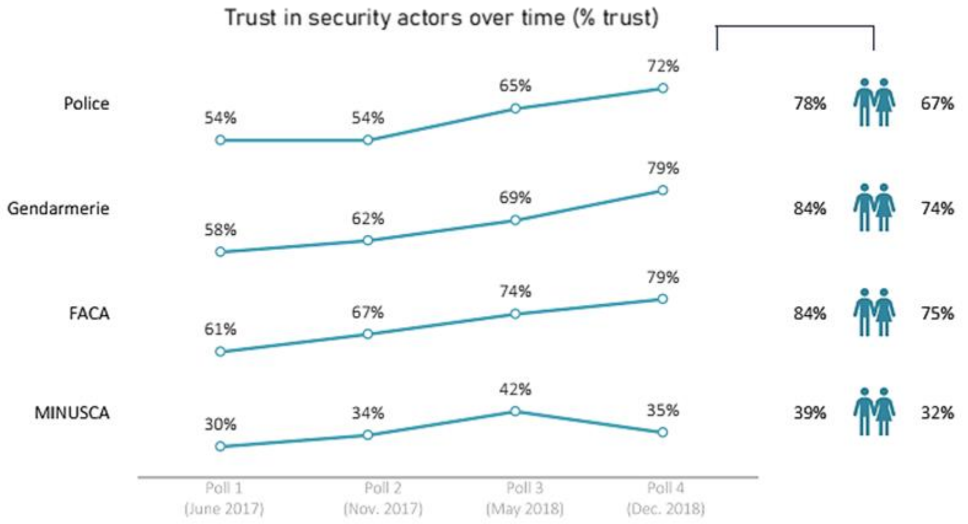


Figure 6. Trust in security actors intervening in the CAR over time [FACA – *Forces armées centrafricaines*]. Results of polls conducted in 12 of the 16 *préfectures* (southeastern provinces were not covered due to insecurity); random 6 336 person sample (50% women). *Source:* (Vinck, Pham, Balthazard, & Magbe, 2019).

Accordingly, the *accuracy* and *completeness* of HUMINT can only increase by either significantly increasing troop strength or by improving in the trust placed in MINUSCA by the host populations. Another option would be to improve the local information and communications technology (ICT) infrastructure, which should increase

[11] Considering *CIA World Factbook* (2020) data on the population (5 990 855, July 2020 estimate) and area (622 984 square kilometers) of the CAR, and the 13 252 uniformed personnel in MINUSCA on March 31, 2020 (MINUSCA Mission Fact Sheet, 2020), MINUSCA can provide 1 uniformed personnel per 452 citizens and per 47 square kilometers. Compare with the same figures for the New York Police Department (NYPD) in 2003 (McGrath, 2006), then the largest police department in the United States: 1 policer officer per 205 city residents and per 0.02 square kilometers (*i.e.*, a *ca.* 140 meter square). If police departments have to resort to hot spots policing because their force density is not enough to be everywhere at once, peace operations like the MINUSCA have little hope of reporting through patrols alone what happens in the host nation territory, potentially creating large uncontrolled areas where rebel groups can thrive.

spontaneous "tipping" by the civilian population by simplifying the process (Berman, Felter, & Shapiro, 2018). Since 2013, in-country efforts by the Emergency Telecommunications Cluster[12] (ETC) aim to do just so, providing internet access and radio programming to the population in 12 major cities (ETC, 2020). However, funding has been sparse (only 8% of the required amount has been raised), mobile network coverage and internet access remain limited, and improvements in ICT do not necessarily translate to increased trust in MINUSCA. Furthermore, poor ICT infrastructure also limits the collection of intelligence through the interception of electronic communications (SIGINT), which is additionally hampered by the lack of specialized personnel and equipment, and by its intrusive nature, which runs against the UN guidelines of non-clandestine information gathering (United Nations, 2019).

Therefore, OSINT and remote sensing, which collect information in a non-intrusive, passive manner, provide the best solutions to complement the limitations of HUMINT in MINUSCA. However, they too face their own shortcomings: first, as explained in the Problem Definition (1.1), imagery alone is not enough to maintain situational awareness in intrastate asymmetric conflicts like the CAR Civil War, and its analysis requires either large numbers of specialized personnel or sophisticated machine vision algorithms. Second, a poor World Press Freedom Index (Reporters Without Borders, 2020) limits the potential of OSINT, though radio and online sources are generally regarded as more reliable than print media. Additionally, regardless of the news sources' bias, the number of relevant events that can be collected from open sources is inferior to that which can be collected directly from the population. (To illustrate, Duursma (2017) (Figure 7) compares the declassified JMAC dataset for the United Nations African Union Mission in Darfur (UNAMID) with the open source-based Armed Conflict Location and Event Data Project (ACLED) dataset regarding armed clash events occurring in the Darfur region between 3 January 2008 and 6 April 2009: of 267 armed clashes, only 20 were reported by ACLED alone, despite it being "the most comprehensive public collection of data on political violence for developing states"). Lastly, even if implemented to complement HUMINT activities, routinely integrating both imagery and OSINT with SAGE might prohibitively tax the workload of JMAC personnel – typically

[12] ETC is "a global network of humanitarian, private sector and government organizations that work together to provide shared communications services in humanitarian emergencies" (ETC, 2020). For the CAR, the network includes the UN Office for the Coordination of Humanitarian Affairs (OCHA), telecommunications company Ericsson, and the government of Luxembourg.

just 10 to 30 persons (Theunens, 2017) –, robbing them of time they need to produce detailed analysis of medium- and long-term trends.

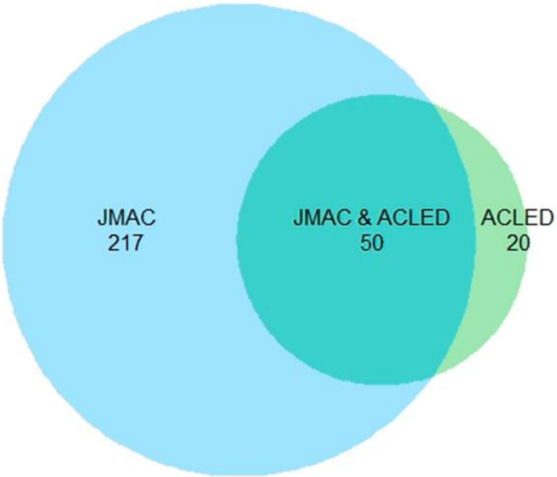


Figure 7. Venn diagram of armed clashes included in the JMAC and ACLED datasets covering the Darfur region between in the Darfur region between 3 January 2008 and 6 April 2009. *Source:* (Duursma, 2017).

Finally, the situational awareness of MINUSCA is also hindered by the way intelligence is processed. Regardless of the source or sensor whence the events came, they are the focus of the intelligence process. However, since reported events tend to make up only a small portion of all conflict-related activity, analyzing their patterns without considering the areas and time intervals where they tend to occur can lead to some misleading conclusions. As Duursma and Karlsrud propose:

> *"Another necessary condition for using the data in SAGE for predictive analyses is to shift from the incident* [relevant event] *as the unit of analysis to a particular geographical area as the unit of analysis (for example, a municipality, a settlement or even a grid cell). This would make it possible to take negative cases (areas where violence is not taking place) into account, which makes it possible to determine what factors drive the onset and termination of armed violence in areas (for example, a peacekeeping deployment), as well as which factors drive the spread of armed violence from one area to another area"* (Duursma & Karlsrud, Predictive Peacekeeping: Strengthening Predictive Analysis in UN Peace Operations, 2019, p. 6)

Considering the processes and limitations described above, efforts to improve the situational awareness of MINUSCA should focus on complementing HUMINT efforts, on reducing the JMAC workload, and on implementing the area/time interval as the unit of analysis. Additionally, since complementing HUMINT with remote sensing and SIGINT would require significant investments in equipment and personnel, relying on OSINT seems the most feasible way forwards.

CHAPTER OUTPUT

In chapter 2, we reviewed the context of our archetypal use case – MINUSCA. To do so, we briefly described the conflict which MINUSCA seeks to solve – the CAR Civil War –, the tasks and goals included in the MINUSCA mandate, the capabilities that MINUSCA possesses to fulfill that mandate, and the processes it uses to maintain situational awareness. We also identified what are the limitations to those processes and advanced possible solutions. In the next chapter, we review the state-of-the-art of spatiotemporal analysis and event extraction in order to select the methods that can best address the limitations to the situational awareness of MINUSCA.

3 Literature Review and State-of-the-Art

CHAPTER GOALS

The goals of Chapter 3 are describing the state-of-the-art of spatiotemporal analysis and event extraction in the context of peacekeeping and law enforcement, and reviewing previous work related to the research problem. It maps to the “Assess Situation” task of the “Business Understanding” phase of CRISP-DM.

3.1 Hot Spots Policing and Spatiotemporal Analysis

“Hot spots policing” – a popular[13] law enforcement tactic – forms the basis of the solution we propose to apply to the research problem. Borrowing the description from then-New York Police Department (NYPD) Deputy Commissioner Jack Maple (1999 *cited by* Braga, Turchan, Papachristos & Hureau, 2019), “the main principle of deployment *[of hot spots policing]* can be expressed in one sentence: ‘map the crime and put the cops where the dots are’”. Its premise is that crime is not spatially homogenous, and that even in administrative divisions with a high event count, the events tend to cluster around specific spatial “hot spots” (Braga, Andresen, & Lawton, 2017). Thus, while law enforcement agencies lack the manpower to be in every street corner in order to completely prevent present crime, future crime can be reduced if those hot spots of past criminal events are identified and addressed, for example, by allocating more patrols to the trouble areas[14]. The effectiveness of the tactic considered proven, being supported by the results of systematic literature reviews conducted by Braga, Turchan, Papachristos & Hureau (2019) and by the Center for Evidence-Based Crime Policy (2020) (Table 3, page 22).

[13] Braga *et al* (2019) recall how a survey conducted on 176 US police departments found that “nearly 9 out of 10 agencies used hot spots policing strategies to deal with violent crime in their jurisdictions”.

[14] One common misconception is that allocating more resources to the trouble areas will simply make the hot spots change location. However, studies have shown that hot spots policing does not lead to spatial displacement of criminal hot spots; rather, it tends to spread crime control to nearby areas (Telep & Weisburd, 2016).

Table 3. Results of systematic literature reviews of studies on the efficacy of hot spots policing.

Systematic Review	Number of studies	Conclusions
(Braga, Turchan, Papachristos, & Hureau, 2019)	65 studies containing 78 tests of hot spots policing interventions.	"Sixty-two of 78 tests of hot spots policing interventions reported noteworthy crime and disorder reductions. The meta-analysis of key reported outcome measures revealed a small statistically significant mean effect size favoring the effects of hot spots policing in reducing crime outcomes at treatment places relative to control places."
(Center for Evidence-Based Crime Policy, 2020)	39 studies, with 25 being considered as successful interventions.	"Over the past two decades, a series of rigorous evaluations have suggested that police can be effective in addressing crime and disorder when they focus in on small units of geography with high rates of crime."

Likewise, echoing the prescription of Duursma and Karlsrud (2019) (2.2.2), conflict-related events in intrastate asymmetric wars like the CAR Civil War are not evenly distributed spatially and temporally. This premise is supported by data and research: Berman, Felter and Shapiro (2018) have analyzed datasets of (georeferenced) conflict-related events pertaining to the wars in Afghanistan (time period: 2005-2015), Iraq (time period: 2005-2012), the Philippines (time period: 1975-2008), and Pakistan (time period: 1988-2010), noticing that the number of events varied greatly across districts and seasons. They concluded that insurgent groups tend to organize locally, so the higher the amount of event data and the better their spatial and temporal precisions, the better will the insurgency be understood. Thus, in the context of a peace operation, the identification of hot spots of conflict-related activity will contribute to the situational awareness of the peacekeeping force, allowing, for instance, changing the spatial distribution of peacekeepers to control emerging crises.

In law enforcement, hot spots of criminal activity are generally discovered through GIS-based methods (Longley, Goodchild, Maguire, & Rhind, 2005) (Smith & Bruce, 2008) (Chainey & Ratcliffe, 2005). These methods generally involve collecting criminal events from patrols, storing them in geodatabases, and using either spatial statistics to discover spatial clusters of events, or using choropleth maps to visually assess the areas with the highest event count. Another possibility is converting the event geodatabase into a space-time cube. Space-time cubes are 3D spaces "consisting of two horizontal dimensions of space (geographic plane) and one vertical dimension of time" (Nakaya & Yano, 2010). As they can be "diced" into smaller bins (*e.g.*, "cubes" with a 15 × 15 km base and a 1-week “height”), they have been used to aggregate geographic events[15] such as criminal activity, allowing the combined study of their temporal and spatial distributions. One recent spatiotemporal analysis technique which relies on space-time cubes and which has been used to model crime (Hashim, Mohd, Sadek, & Dimyati, 2019) is ESRI’s Emerging Hot Spots Analysis. Emerging Hot Spots Analysis uses space-time cubes generated from event datasets as input, computing the Getis-Ord Gi* statistic (Getis & Ord, 1992) of each bin and the Mann-Kendall Trend Test (Mann, 1945) (Kendall, 1975) of each cube column (time series of bins) to discover statistically significant spatial hot spots of events and their associated temporal trends. The output 2- and 3-dimensional visualizations display the detected spatiotemporal hot spots, allowing decision-makers to identify new hot spots, to track seasonal fluctuations, and to gauge the effects of the responses directed at the known hot spots.

In peace operations, GIS is also widely regarded as a crucial tool for storing and visualizing conflict-related events. The Digital Toolkit report made by the United Nations Department of Political and Peacebuilding Affairs and the Centre for Humanitarian Dialogue (2019) describes several proprietary UN GIS platforms primarily used to manipulate imagery but also mentions two open-source initiatives which map conflict-related events using web GIS: Liveuamap[16] and Ushahidi[17]. Ushahidi is a web platform which compilates and maps crowdsourced data, *i.e.*, reports sent by the populations of crisis areas describing relevant events (Manning, 2018), while Liveuamap (2020) uses “AI Web crawlers” to scrape possible conflict-related news reports from online sources and a team of analysts and editors to convert the scraping output into events, displaying the result using web GIS. Neither of the platforms performs spatiotemporal analysis on the collected events. The sole mention of spatiotemporal analysis in the reviewed UN literature happens in the JMAC handbook (Martin-Brûlé & Assouli, 2018), which describes the practice of creating hot spots maps as “risk mapping”, (*i.e.*, creating graphic representations of risk distribution), and risk maps as one of the typical outputs of the JMAC. The handbook also mentions the ArcGIS software and add-ons to the *IBM i2* as the tools used to create risk maps but does not describe the specific techniques used in their creation, nor details any performance benchmarks for hot spots analysis in peace operations.

[15] When studying events at a “geographic” scale, their *z*-coordinate can be safely disregarded, as knowing the precise spatial location of each event is often less important than understanding the pattern of similar events.

[16] https://liveuamap.com/

[17] https://www.ushahidi.com/

3.2 Event Extraction

To “feed” the spatiotemporal analysis of conflict-related events, we propose to extract events from online news sources. Event extraction can be understood as the process of parsing unstructured text data with natural language understanding engines with the goal of detecting events and extracting their arguments, *i.e.*, entities which describe their dimensions (*e.g.*, participants, place, time) (Wei & Bang, 2019). It can be closed-domain or open-domain. In closed-domain event extraction, the relationships between entities are encoded in a predefined *schema* or *frame*, for instance, an “attack” event with the “attack type”, “attacker”, “target”, “location”, and “date” arguments. Conversely, on open-domain event extraction, entities belonging to the same event are clustered together, but their relationship to the event is not explicitly defined, only detected.

The state-of-the-art of event extraction (Wei & Bang, 2019) can be divided into two main approaches: systems based on pattern matching and systems based on machine learning. At its core, pattern matching systems detect events whenever candidate entities (n-grams) in the input text data match predefined event frames. For instance, if a simple “attack” event frame were defined as “[*Agent*] attacks [*Target*] in [*Location*]”, with “attack” being the event trigger, the string “FACA attacks UPC in Bouar” would be extracted as an attack event. As expected, this is a high precision, poor recall approach: since triggers and frames must be manually predicted during development, events with equivalent but unforeseen triggers will not be detected. Therefore, state-of-the-art pattern matching systems define patterns of parts-of-speech (POS) and use other heuristics (*e.g.*, for a given argument, the most frequent candidate is often the most relevant) to extract events.

One state-of-the-art example of the pattern matching approach is the “Giveme5WH1” system (Hamborg, Breitinger, & Gipp, 2019), whose goal is to extract the main event of news articles by extracting the answers to the 5WH1 questions (“Who?”, “What?”, “When?”, “Where?”, “Why?”, and “How?”). Giveme5WH1 uses a set of four independent “Phrase Extraction” chains and six “Candidate Scoring” functions to extract event arguments (Figure 8, page 25), performing event extraction in three steps:

- First, after preprocessing the input text data and converting all named entities into their canonical form (*i.e.*, dates into timestamps, locations into their geographic coordinates), the Phrase Extraction chains select candidate arguments using encoded rules and heuristics. For instance, for the “Who” and “What” arguments, each noun phrase (NP) – verb phrase (VP) pair (subject and object) which is a direct child of the sentence (*i.e.*, not a relative clause) is considered a candidate for the "Who"-"What" pair of arguments, since the subject of a sentence is usually the "Who", and the predicate the "What". For the "Where" argument, all geographic coordinates are considered as candidates.

- Then, candidate entities are scored according to heuristic factors, such as their position and frequency in the input text, *e.g.*, entities which appear earlier (*i.e.*, in the headline and in the lead) and more often are likely the arguments of the main event reported by the input news article. Position and frequency are criteria common to all arguments, but there are argument-specific factors as well: for instance, the more specific (*i.e.*, whose area or type is smallest) the candidate locations, the better their score, since larger locations are often mentioned to provide context for the smallest.
- Finally, the highest scoring candidates are extracted as the event arguments.

Giveme5WH1 was evaluated using 120 news articles annotated by three human researchers, achieving a mean average generalized precision (MAgP) of 0.73 for 5WH1 and 0.82 for the first 4W ("Who", "What", "Where", "When") (see Table 4, page 29, for an indirect comparison with machine learning event extraction systems).

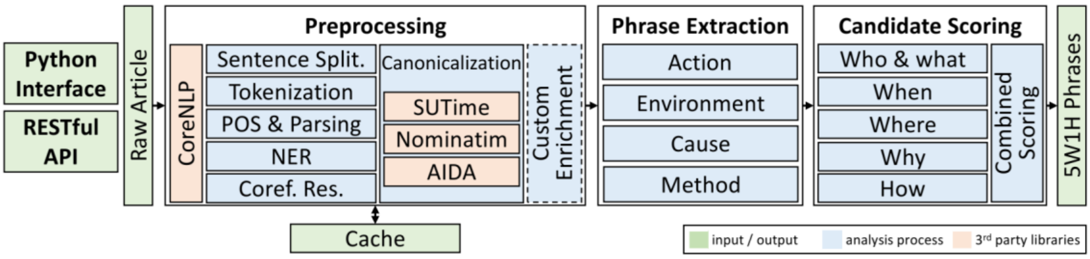


Figure 8. Structure of the Giveme5WH1 pipeline. *Source:* (Hamborg, Breitinger, & Gipp, 2019).

As for the machine learning event extraction systems, they can be divided into supervised and unsupervised learning systems according to the classification/clustering model they use (Wei & Bang, 2019). Unsupervised learning event extraction systems rely on models such as Latent Dirichlet Allocation (LDA) or Self-Organizing Maps (SOM) to cluster word vectors in events or document vectors into topics and are mostly used for open-domain event extraction, since clusters of entities describing the same event are created without regard to labeled training data or event frames. In contrast, supervised learning event extraction systems use models such as Support Vector Machines (SVM), Neural Networks, or Markov Logic Networks (MLN), which require training but can classify the input according to predefined labels, making them more suitable for closed-domain event extraction. In such models, words in the labeled dataset are embedded into vectors alongside their POS tags, entity type, and other contextual information provided by using pre-trained word embedding models like Word2Vec[18]. Then, they are used to train the models, often by using the error between the predicted label and the true label to influence the internal parameters of the model (*e.g.*, neuron weights through backpropagation) that determine which label is assigned to each input word vector.

To illustrate the state-of-the-art of supervised learning event extraction models, we review two "Joint Event Extraction" systems: a system based on Graph Convolutional Neural Networks (GCNN) (Liu, Luo, & Huang, 2018) and a system that uses a local Linear Regression with Maximum Likelihood Estimation model and a Markov Logic Networks (LR/MLN) to fill arguments of event frames defined by the FrameNet Corpus[19] (Li, Cheng, He, Wang, & Jin, 2019). They are "Joint" systems in the sense that their goal to extract events from unstructured text data while "jointly" using the internal relations between events of the same document to refine overall results (Figure 9 illustrates the challenge). The system by Liu, Luo, & Huang (2018) achieves this by building a GCNN in which the nodes of the graph layer are all the unique word-vectors in the training data, while their edges are their relationships with other nodes according to the ACE2005[20] event frames (Figure 10, page 27). Conversely, the system by Li, Cheng, He, Wang, & Jin (2019) uses the FrameNet Corpus to define event frames and their relations (Figure 11, page 28), using a local Linear Regression with Maximum Likelihood Estimation model to extract the event trigger and arguments and a MLN to ensure that their values are consistent with the hierarchical relations between events. Table 4 (page 29) summarizes training, testing, and results of the GCNN and LR/MNL systems alongside other event extraction systems presented in this chapter in order to provide performance benchmarks for the project.

[18] Word2Vec (Mikolov, Chen, Corrado, & Dean, 2013) is an algorithm which represents words as vectors based on their context and meaning. It consists of a two-layer neural network trained with one or more linguistic corpora which yields the most likely word given the surrounding words (Continuous Bag-of-Words model) or the most likely surrounding words given a single input word (Continuous Skip-gram model).

[19] The FrameNet Corpus (The FrameNet Project, 2020) is a database of frames, frame objects (arguments), and lexical units (instances of arguments) extracted from more than 200 000 annotated sentences.

[20] The ACE2005 event corpus (Walker, Strassel, Medero, & Maeda, 2006) is an annotated event dataset often used to train and evaluate event extraction systems. It defines 8 event types and 33 subtypes, with each event subtype corresponding to a set of argument roles (Wei & Bang, 2019).

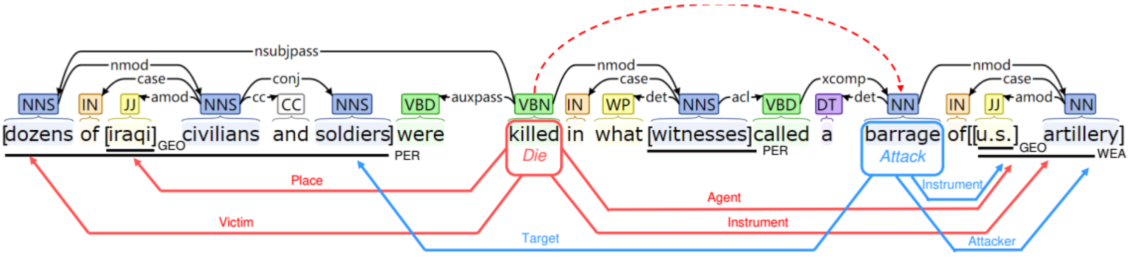


Figure 9. Example of a string containing two related events. Considering that the "barrage" event *caused* the "killed" event allows the event extraction system to identify the "target" of the "barrage" event and the "agent" of the "killed" event. *Source:* (Liu, Luo, & Huang, 2018).

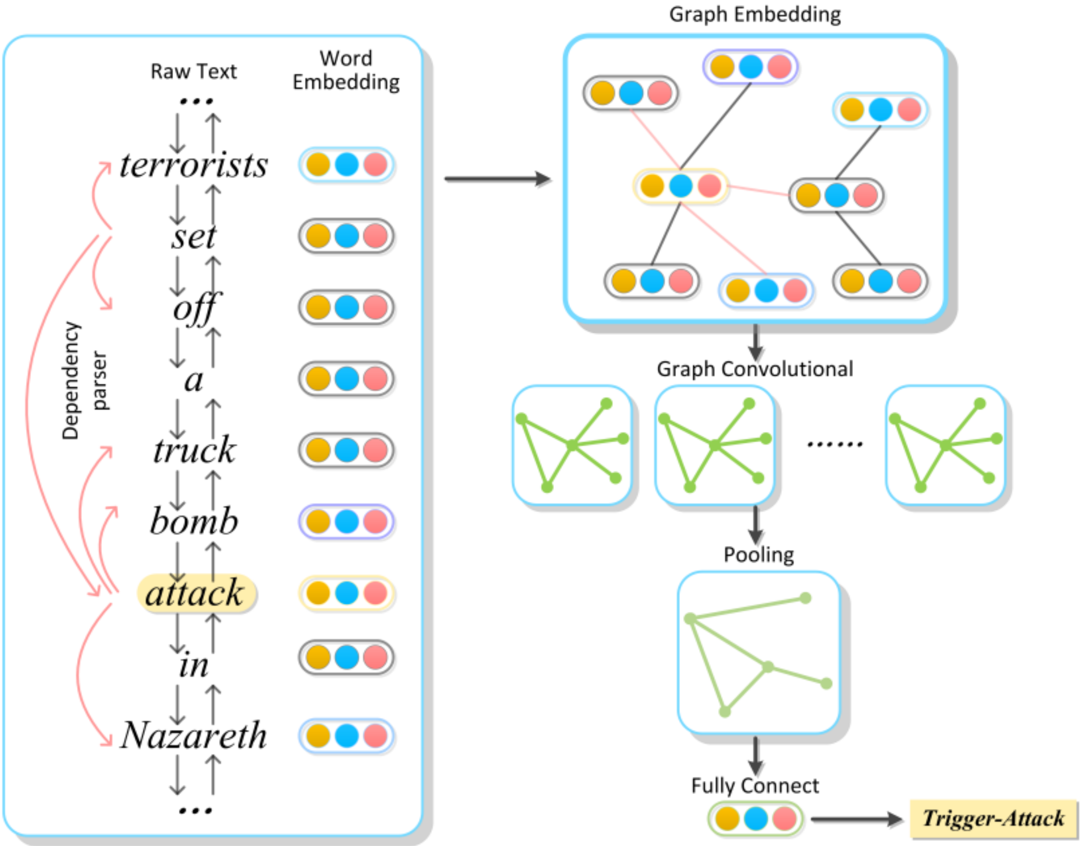


Figure 10. Diagram of a CGNN used to extract an event trigger. *Source:* (Wei & Bang, 2019).

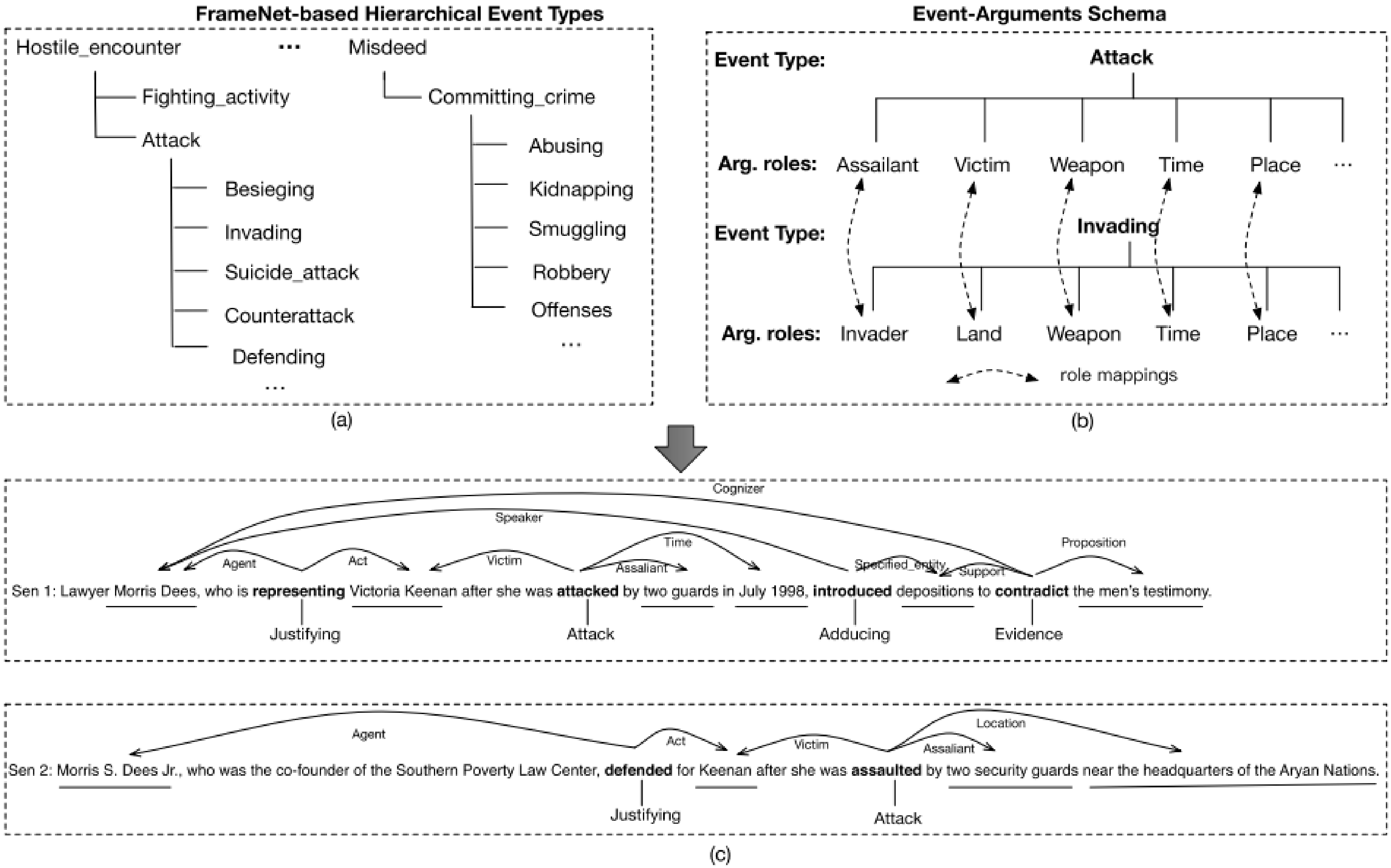


Figure 11. Diagrams of the hierarchical relations between event frames established by the FrameNet Corpus and leveraged by the LR/MLN approach. Inset (b) describes the mapping between arguments of different events. *Source:* (Li, Cheng, He, Wang, & Jin, 2019).

Table 4. Indirect comparison between state-of-the-art event extraction/NLU systems reviewed in Section 3.2. No methods can be directly compared with the others, since their testing procedures, datasets and even performance metrics differ. Note that P refers to precision, R to recall and F1 to the F1-score metric (except for "Giveme5W1H", when it refers to Mean Average Generalized Precision). Also note that "overall" performance is the average of the "event detection" and "argument extraction" metrics.

	Training set	Test set		OVERALL			EVENT DETECTION			ARGUMENT EXTRACTION		
				P	R	F1	P	R	F1	P	R	F1
Snips NLU	70	100	Proprietary	0.80	0.77	0.79	*No data*					
	2000	100		0.95	0.92	0.93						
GCNN	21 090	881	ACE 2005	0.76	0.69	0.72	0.80	0.72	0.76	0.71	0.66	0.68
LR/MLN	88 989	2 472	Proprietary	0.80	0.63	0.70	0.86	0.72	0.78	0.75	0.54	0.63
Giveme5 WH1	100	120	Proprietary	*No data*		*0.73*	*No data*					

Finally, Natural Language Understanding (NLU) engines, such as the engines that power voice assistants and chatbots, can also be considered as part of the state-of-the-art of event extraction, as they tackle similar problems: while an event extractor detects events and extracts arguments in the input text data, the NLU engine of a chatbot must detect the intent (*e.g.*, a question or an order) and the intent entities (arguments) in the input data and respond accordingly. One open-source, state-of-the-art NLU engine is Snips NLU (Coucke, et al., 2018), which combines deterministic and probabilistic "intent parsers" to detect intents (events) and their entities (Figure 12, page 30). The deterministic intent parser is equivalent to a pattern matching event extractor, using the training "utterances" (annotated sentences containing the relevant intent) to define intent frames in which only the specific entities can vary (*e.g.*, the question "Will it rain tomorrow in [*Location*]?"). The entities can be built-in or custom, and their possible values come not only from the training utterances, but also from lists of possible entity values defined by the users, allowing the input of expert knowledge beyond training utterances. The probabilistic intent parser uses Conditional Random Fields (CRF) trained on the training utterances (vectorized using Word2Vec) to fill the intent entity "slots" based on the probability that a certain word maps to a certain entity given the surrounding words. Then, it computes the confidence level of the input string being an intent with a Logistic Regression model. Thus, for each input string, Snips NLU tries to extract intents using deterministic intent parser first, since its specificity (precision) is high, deploying the probabilistic intent parser only when the deterministic parser yields no results. Therefore, Snips NLU can be described as a "hybrid" NLU system, since it combines the strengths of pattern matching and machine learning. As for the other event extraction systems, Table 4 (page 29) summarizes training, testing, and results of Snips NLU.

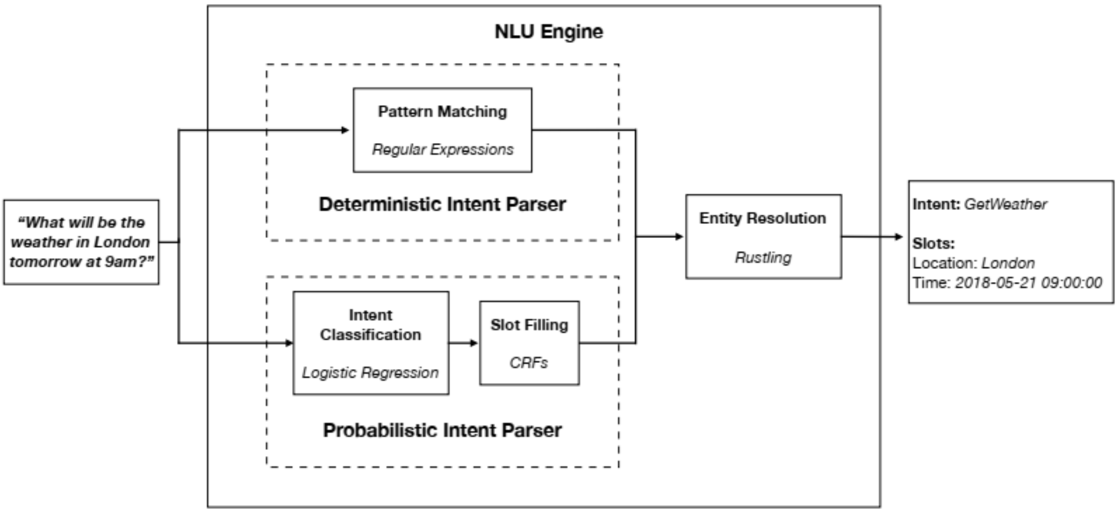


Figure 12. Structure of the Snips NLU Engine pipeline. "Entity resolution" describes the process of converting extracted entities to their canonical forms. *Source:* (Coucke, et al., 2018).

3.3 Related Work

To gauge related work on the research problem, we searched for project-related keywords in the title, abstract and keywords of peer-reviewed publications in the Scopus[21] and Web of Science[22] databases. Table 5 (page 31) presents the number of results per query, showing that despite the large number of results for queries containing individual or equivalent keywords (*e.g.*, "peacekeeping" OR "peace operations"), very few peer-revied publications were found when using the intersection of two keywords, while the intersection of three keywords produced no results at all. This is consistent with the emerging nature of the technologies used by the proposed solution (Figure 13, page 32) and with the project context, as the access to research and development projects dealing with OSINT and peace operations is usually restricted to active-duty members of military or peacekeeping organizations.

[21] https://www.scopus.com/

[22] http://apps.webofknowledge.com

Table 5. Number of results (peer-reviewed publications) found in the Scopus and Web of Science databases per query (2000-2020).

Query Type	Query	Scopus	Web of Science
Project keywords	"event extraction"	1 024	511
	"natural language understanding"	2 172	1 136
	"spatiotemporal analysis" OR "spatiotemporal data analysis"	57 264	2 030
	"peacekeeping" OR "peace operations"	5 543	2 755
	"open-source intelligence" OR "OSINT"	405	219
Domain keywords	"text mining"	18 157	11 865
	"natural language processing"	71 297	19 719
	"GIS" OR "geographic information systems"	211 768	85 196
Combinations of project keywords	"event extraction" AND ("spatiotemporal analysis" OR "spatiotemporal data analysis")	**1**	0
	"event extraction" AND ("peacekeeping" OR "peace operations")	0	0
	"event extraction" AND ("open-source intelligence" OR "OSINT")	**9**	0
	"natural language understanding" AND ("spatiotemporal analysis" OR "spatiotemporal data analysis")	0	0
	"natural language understanding" AND ("peacekeeping" OR "peace operations")	0	0
	"natural language understanding" AND ("open-source intelligence" OR "OSINT")	0	0
	("spatiotemporal analysis" OR "spatiotemporal data analysis") AND ("peacekeeping" OR "peace operations")	0	0
	("spatiotemporal analysis" OR "spatiotemporal data analysis") AND ("open-source intelligence" OR "OSINT")	0	0
	("peacekeeping" OR "peace operations") AND ("open-source intelligence" OR "OSINT")	**1**	0
	("event extraction") AND ("spatiotemporal analysis" OR "spatiotemporal data analysis") AND ("open-source intelligence" OR "OSINT")	0	0
	("event extraction") AND ("spatiotemporal analysis" OR "spatiotemporal data analysis") AND ("peacekeeping" OR "peace operations")	0	0
	("natural language understanding") AND ("spatiotemporal analysis" OR "spatiotemporal data analysis") AND ("open-source intelligence" OR "OSINT")	0	0
	("natural language understanding ") AND ("spatiotemporal analysis" OR "spatiotemporal data analysis") AND ("peacekeeping" OR "peace operations")	0	0

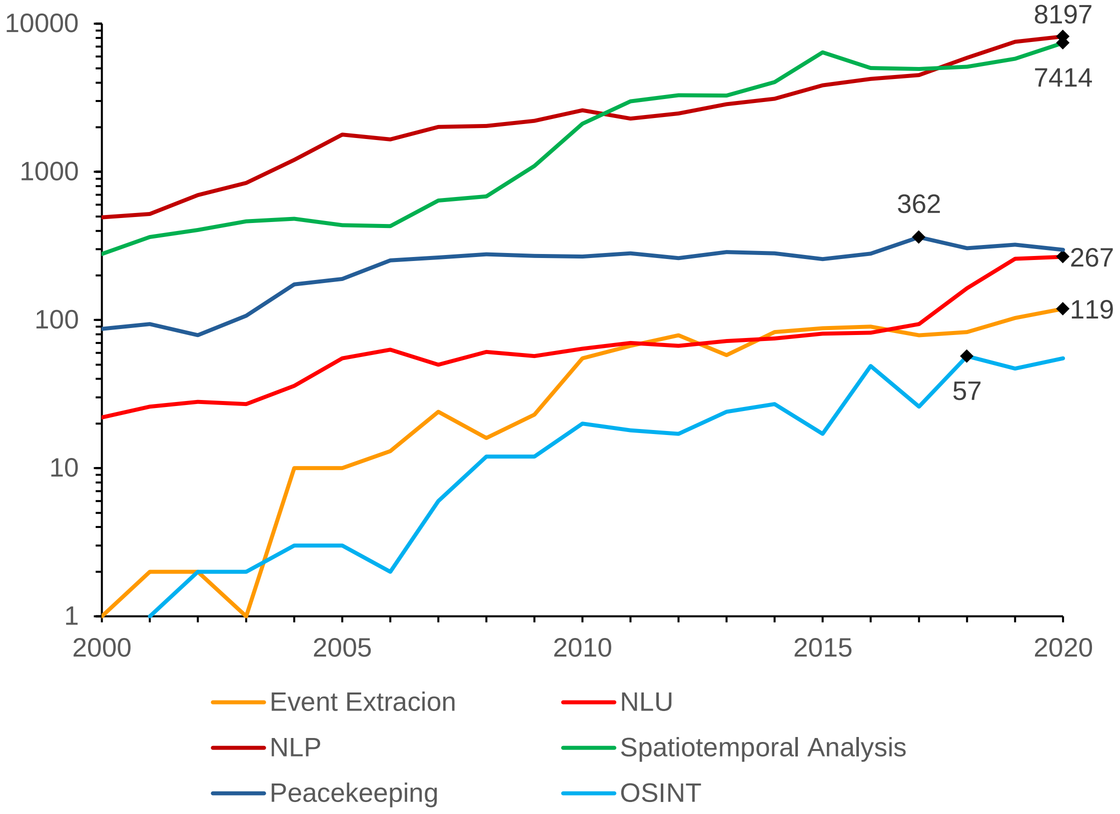


Figure 13. Number of peer-reviewed articles in the Scopus database containing the project keywords in the title, abstract and keywords, per year, 2000-2020. NLP refers to Natural Language Processing; we included it to illustrate the increasing amount of scientific research on the topic.

However, three relevant projects were found amongst the limited number of Scopus results. First, Farnaghi, Ghaemi, & Mansourian (2020) present the Dynamic Spatio-Temporal Tweet Mining (DSTTM) system, an open-domain machine learning event extraction system with the goal of detecting events by clustering geotagged tweets. Tweets are clustered according to their spatial, temporal, and semantic distances using the non-parametric Ordering Points To Identify the Clustering Structure (OPTICS) algorithm, an evolution of the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm which takes into account the spatial heterogeneity of the study area (*i.e.*, differences in tweet density between regions). DSSTM was tested on an unspecified number of tweets posted in the North Carolina-South Carolina region between shortly before the landfall of Hurricane Florence (12 September 2018) and shortly after its demise (19 September 2018). No ground truth dataset exists, so DSTTM performance was assessed by feeding the same test to a system using DBSCAN and comparing its results with those of DSTTM. Overall, though the DSTTM system performed well at clustering tweets into events and at extracting their keywords using the Hierarchic Dirichlet Process, the lack of event frames excludes the explicit extraction of event arguments, while the reliance on geotagged tweets presupposes that each tweet is a local report. Both these factors limit the importance of this study for the present project.

The second work, “CrisMap” (Avvenuti, Cresci, Del Vigna, Fagni, & Tesconi, 2018), is a system which also analyzes tweets with the purpose of detecting events – specifically, it aims to detect damage inflicted by natural disasters and the affected municipalities. Rather than performing open-domain event extraction on geotagged tweets from a relevant study area, like DSTTM, CrisMap performs *de facto* closed-domain event extraction by considering a single “Damage” event with “Location” as its sole argument. It uses SVMs to perform two-step binary classification on the input tweets, classifying them as “Relevant/Irrelevant” according to whether they are related to the natural disaster and “With Damage/Without Damage” according to whether they contain a “Damage” event. Then, CrisMap “geoparses” the tweets that are labelled as “Relevant” and “With Damage”, *i.e.*, extracts their “Location” argument and geocodes the result (*i.e.*, enriches it with geographic coordinates and other geographic information). Tweets that are “Relevant” and “With Damage” and that were successfully geoparsed are then stored in a database and visualized in choropleth maps using the Kibana[23] platform. CrisMap was tested on datasets covering 5 natural disasters in Italy between 2009 and 2016, for a total of 15 825 tweets. Regarding event detection, it achieved an F1 of 0.82 for “Relevant” tweets and an F1 of 0.83 for “With Damage” tweets. In geoparsing, it achieved an F1 of 0.84. As for its ability to detect the municipalities struck by the disasters in the datasets, the results of CrispMap for the 2012 earthquake in the Emilia Romagna region and the 2013 Sardinia flash floods were compared with official damage assessment maps produced after the disasters. In the Emilia earthquake, CrisMap achieved an F-Measure of 0.303 the detection of all affected municipalities, but an F-Measure of 0.897 for the detection of the municipalities that suffered significant damage. The same pattern occurs in the Sardinia flash floods: a F-Measure of 0.222 for all affected municipalities and 0.667 for the ones that suffered significant damage. Again, though CrisMap performs well its intended purpose and exemplifies the use of binary classification for event detection and geoparsing/geocoding to discover geographic information of event locations, its importance to the present project is limited, as it focuses on detecting “Damage” events during high-magnitude disasters, while the present project aims to continuously detect low-magnitude conflict-related events perpetrated by a plethora of

[23] https://www.elastic.co/products/kibana

factions (recall Table 2). Furthermore, CrisMap conducts no spatiotemporal analysis beyond choropleth maps.

Finally, Kotzé, Senekal, & Daelemans (2020) experiment with various text classification algorithms to detect reports of violent events in WhatsApp groups. Specifically, they attempt to use SVM, Logistic Regression, Random Forest, and Gradient Boosting classifiers to label messages vectorized using the Term Frequency–Inverse Document Frequency (TF-IDF) (Spärck Jones, 1972) and Word2Vec processes according to five event types: "Land grabs", "Farm attacks", "Crime", "Protests", and "Safe" (irrelevant messages). To create the training/evaluation datasets, they collected 8 398 unique messages between 30 May 2018 and 18 February 2019 from 15 English and Afrikaans WhatsApp groups that focus on reporting crime and violent events in South Africa. Then, two annotators manually classified each message according to the type of event it contained, with 46.91% containing reports on violent events. Since the five event types create a multiclass classification problem, and since some event types constitute a very small portion of the annotated dataset (*e.g.*, only 0.63% of the messages contain "Land grabs"), oversampling was used to mitigate class imbalance during training, which was conducted using an 80% training split of the annotated dataset. The best result was obtained by the Logistic Regression model using TF-IDF vectorization, which achieved a F1-score of 0.789 and an accuracy of 0.899. However, despite the noteworthy results, the research is also of limited utility to the present project, as it lacks the spatiotemporal analysis component altogether. Nevertheless, it is also the closest to the present project in terms of event detection/extraction goals, even if argument extraction is not attempted at all.

Having reviewed related work found on the Scopus and Web of Science databases, we conclude that none combines "full" event extraction (event detection and argument extraction) with emerging hot spots analysis – much less in the context of peace operations –, leaving us without a "model project" to orient the development of our own. However, insights garnered from the architecture of their event extraction and spatial analysis pipelines and their strategies to train classification models while mitigating class imbalance, to cluster tweets describing the same real-word event, and to evaluate the results without ground truth datasets are incorporated in the development of the proposed solution, ORÁCULO, along the methods for event extraction and spatiotemporal analysis reviewed in the sections 3.1 and 3.2.

Chapter Output

In chapter 3, we reviewed the state-of-the-art of both spatiotemporal analysis and event extraction, as well as recent work related to the research problem, in order to identify how to best address the limitations to the situational awareness of MINUSCA identified in chapter 2. In the next chapter, we describe what state-of-the-art event extraction and spatiotemporal analysis methods we selected to address those limitations, how are these methods implemented in a prototype – ORÁCULO –, and what data is used to develop and test that prototype.

4 System Architecture and Implementation

CHAPTER GOALS

The goals of Chapter 4 are describing the architecture, training and evaluation data, and structure and execution of the proposed solution to the research problem, ORÁCULO. It maps to the "Determine Data Mining Goals" and "Produce Business Plan" tasks of the "Business Understanding" phase of CRISP-DM (4.1), but also to its "Data Exploration" (4.2), "Data Preparation" (4.3 and 4.4) and "Modelling" (4.5) phases.

4.1 System Architecture

The problem definition (1.1), requirements and constraints (2.2), and state-of-the-art (Chapter 3) guided the development of our proposed solution, ORÁCULO, a closed-domain supervised event extraction system that detects significant spatiotemporal hot spots ("hot spots") of conflict-related events ("events") extracted from online news sources. It detects a single event type, the conflict-related event, whose arguments are *action*, *agent*, *target*, *date*, *location*, and *effects* (Table 6).

Table 6. Arguments of the conflict-related event in ORÁCULO.

Name	Associated questions	Precision	Example
Action	*What* happened in the *event*?	Verb/description	"repels attack"
Agent	*Who* executed the *event*?	Faction	"FACA"
Target	*Who* suffered the *event*?	Faction	"UPC"
Date	*When* did the event happen?	Day	"20 May 2020"
Location	*Where* did the *event* happen?	City/town	"Obo"
Effects	*What* are the immediate effects of the *event*?	# casualties	"10 killed"

To detect hot spots of events extracted from news sources, ORÁCULO comprises an independent function and two components, receiving human input at two moments:

- The analyst selects promising online news sources.
- The scraper function scrapes, translates, and preprocesses news from the online sources selected by the analyst and stores them in a scraped news dataset.
- The **Event Extractor** extracts events from the scraped news dataset, discovers their locations (geocoding), and merges extracted events if they refer to the same real-world incident, generating and refining the event dataset. Its output is the *merged event dataset*.

- The analyst supervises the *merged event dataset*, discarding false positives and generating a *refined event dataset*.
- The **Hot Spots Detector** generates a geodatabase and a space-time cube from the *refined event dataset*, performing emerging hot spots analysis on the cube to detect event hot spots. Its outputs are the *event geodatabase* and the *event space-time cube*.

In turn, to perform their allotted tasks, the main components comprise several functions, which we summarize in the structure chart of ORÁCULO (Figure 14, page 38) and describe in detail in sections 4.3 and 4.5. The system and its components were implemented using the Python programming language and the libraries described in Table 7, while its outputs can be visualized and manipulated using the ArcGIS Pro GIS.

Table 7. Python libraries, Web Services, Gazetteers and Contextual Information used by the functions of ORÁCULO.

Component	**Function**	**Python Library, Software, Web Service, Gazetteer, Contextual Information** (Source) *(purpose)*
(general purpose)	*(general purpose)*	Pandas (2020) *(data manipulation)*
		Re (2020) *(regex – text manipulation)*
		ArcGIS Pro (ESRI, 2020c) *(spatial data manipulation and visualization)*
Event Extractor	scraper	GetOldTweets3 (Henrique, 2020) *(scraping)*
		NLTK (2020) *(text preprocessing)*
		TextBlob (Loria, 2020) *(translation)*
	snips NLU event extractor	Snips NLU Engine (Ball & Doumouro, 2020) *(extraction)*
		NATO Mission Task Verbs (APP-6(C), 2011) *(gazetteer)*
		CAR settlements dataset (OCHA, 2018) *(gazetteer)*
		sklearn (Pedregosa, et al., 2011) *(evaluation)*
	geocoder	geocoder (Carriere, 2020) *(geocoding)*
		GeoNames (2020) *(geocoding web service)*
		CAR settlements dataset (OCHA, 2018) *(geocoding gazetteer)*
		metaphone (Collins, 2016) *(toponymy matching with the double metaphone algorithm)*
	event merger	Pandas (2020) *(aggregation)*
Hot Spots Detector	geodatabase generator	CAR administrative regions shapefile (SOGEFI Ingénierie Géomatique, 2018) *(contextual information)*
		CAR settlements shapefile (OCHA, 2018) *(contextual information)*
	emerging hot spots detector	statsmodels (Skipper & Perktold, 2010) *(temporal autocorrelation)*
		matplotlib (Hunter, 2007) *(visualization)*
		ArcPy (ESRI, 2020d) *(emerging hot spots analysis, spatial autocorrelation)*
		xarray (Hoyer, Kleeman, & Brevdo, 2020) *(space-time cube manipulation)*

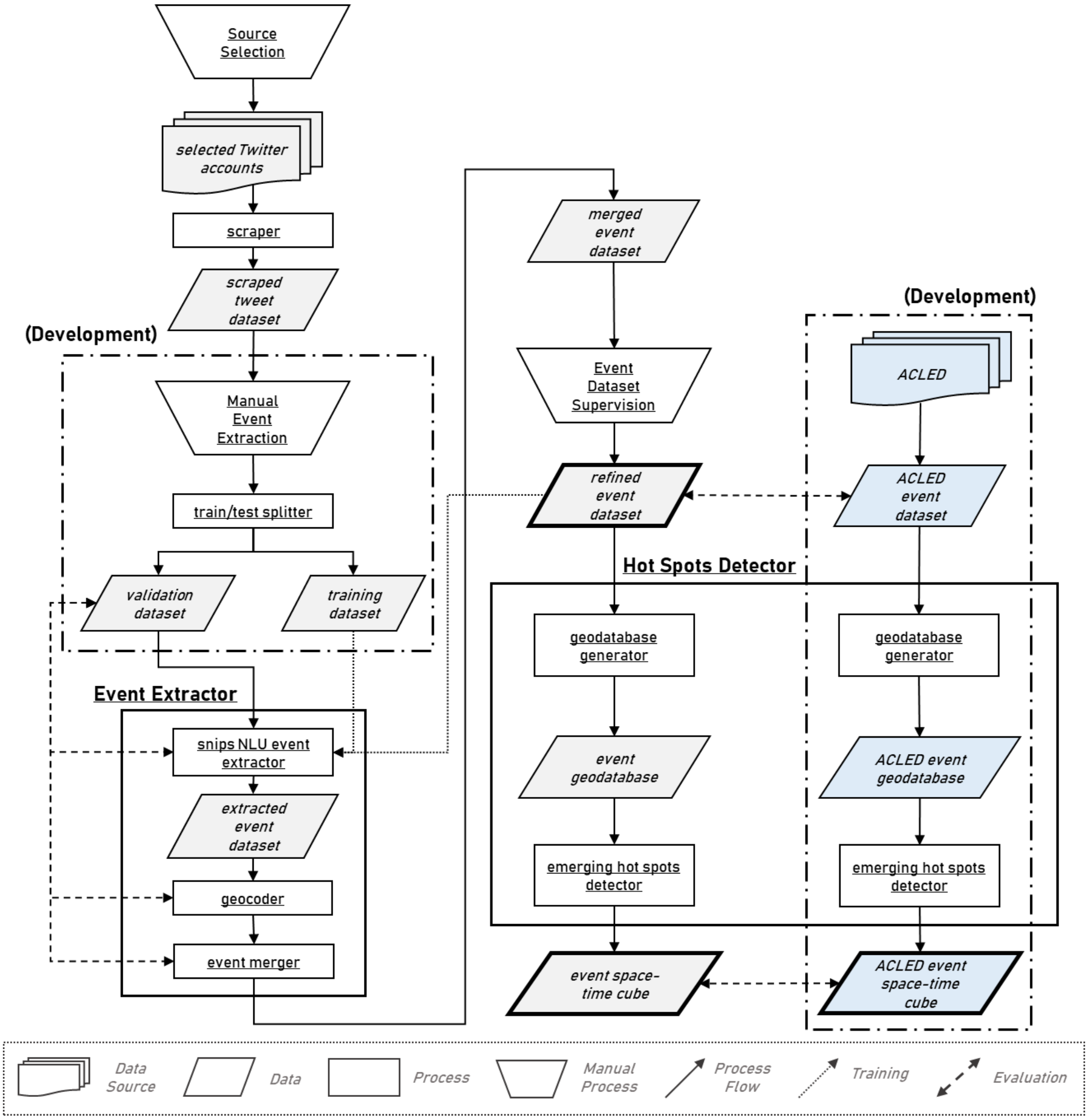


Figure 14. Structure chart of ORÁCULO. The reasoning behind the selection of Twitter and ACLED as data sources is explained in 4.2. Also note how the processes and data surrounded by a dash-dotted line are conducted for development and evaluation purposes only: during normal operation, the *scraped tweet dataset* is fed directly to the snips NLU event extractor, and no ACLED datasets are generated.

4.2 Data and Preprocessing

The base data necessary to develop and test ORÁCULO consists of the *scraped tweet dataset*, a dataset of tweets scraped from the Twitter accounts of selected news organizations that cover the CAR Civil War. Additionally, the *ACLED event dataset* was retrieved from ACLED through their Application Programming Interface (API) to serve as *near-ground truth* during the evaluation process. Both datasets cover the period between 06 June 2019 and 26 June 2020 (the "scraping interval").

4.2.1 ACLED Event Dataset

The *ACLED event dataset* contains 222 CAR Civil War-related events that were manually extracted from public sources by ACLED, a geographically dispersed network of researchers (Raleigh, Linke, Hegre, & Karlsen, 2010). ACLED codes their events (*i.e.*, extracts their arguments) according to a proprietary system (Table 8, page 40), meaning that the *ACLED event dataset* already consists of structured data: each row of the dataset describes a unique event (incident).

We consider the *ACLED event dataset* as "near-ground truth" for three reasons: first, the ideal choice – the (presumed) SAGE event dataset – is controlled by the MINUSCA JOC and is not made public; second, ACLED is considered as the most complete public data repository about conflict in developing countries (recall 2.2.2); third, their own assessment of event coding precision in the *ACLED event dataset* describes most events as coded with the best geographic and temporal precisions (Figure 15, page 41). Consequently, despite ACLED not being a primary data source for ORÁCULO, we collected its data before scraping online news, as the exploration of the *ACLED event dataset* led to the discovery of a set of news sources that reliably reported CAR Civil War events. In turn, this guided the selection of the most promising news sources for ORÁCULO. Figure 15 (page 41) describes the most prolific sources in the *ACLED event dataset*.

Table 8. Selected event arguments/data columns of the ACLED coding system. *Adapted from:* (ACLED, 2019).

Data Column	Description	Unique Values	Notes
event id	An individual numeric identifier (updated annually).	222	Unique ID for each event; *de facto* primary key. If two or more events happen between the same actors, in the same day, and in the same location, ACLED aggregates them in a single event.
event date	The day, month, and year on which an event took place.	156	Each day of a multi-day event is coded as a single event in the database. If no precise date is found, events are coded as happening in the midpoint of the available week/month.
time precision	A numeric code indicating the level of certainty of the date coded for the event.	3	1 (highest, specific day) to 3 (lowest, month).
event type	The type of event.	6	Either Battles, Explosions/Remote, Violence, Violence against civilians, Protests, Riots, or Strategic developments
actor1	The named actor involved in the event.	33	One-sided events have only actor1 (there is no agent or target).
actor2	The named actor involved in the event.	30	
country	The country in which the event took place.	1	Redundant in the *ACLED event dataset*, since it covers only the CAR
admin1	The largest sub-national administrative region in which the event took place.	17	*Préfectures*.
location	The location in which the event took place.	70	If no precise location is found, location is the name of the capital of the reported region.
latitude	The latitude of the location.	72	In the WGS84 Geographic Coordinate System (GCS) (EPSG:4326).
longitude	The longitude of the location.	71	
geo precision	A numeric code indicating the level of certainty of the location coded for the event.	3	1 (highest, specific town) to 3 (lowest, region).
source	The source of the event report.	19	
source scale	The scale of the source.	4	Either local, regional, national, or international
notes	A short description of the event.	222	Usually a short description of the event.
timestamp	The UNIX timestamp this data entry was last updated	77	Integer data type.
fatalities	The number of reported fatalities which occurred during the event.		When multiple estimates exist, the lowest estimate is recorded. Integer data type.

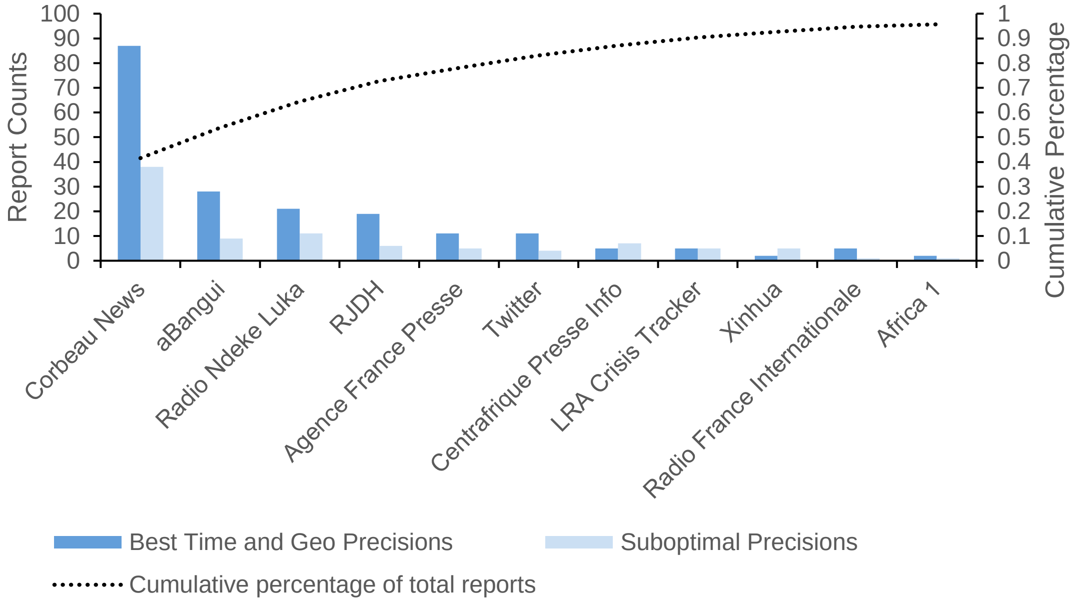


Figure 15. Report counts by source and temporal/geographic precisions in the *ACLED event dataset* (the same event can be extracted from multiple reports). The graph displays only the sources responsible for 288 reports – 96% of the 301 reports that generated the 222 events in the *ACLED event dataset*. It is unclear whether "Twitter" refers to the Twitter accounts of other sources or to the accounts of specific users that reported an event. "Best Time and Geo Precisions" refers to reports that led to ACLED events with a value of 1 (best) in the *time precision* and *geo precision* features.

4.2.2 Scraped Tweet Dataset, Preprocessing and Training/Validation Dataset

The acquisition of unstructured data (news reports) from online news sources ("scraping") began after identifying the most prolific sources of the *ACLED event dataset* (Figure 15). We chose to scrape tweets from the active and Africa-specific Twitter accounts of the most prolific sources because as a platform from which to scrape news reports, Twitter holds several advantages that simplified the development of the prototype of ORÁCULO. First, each tweet is limited to 240 characters, forcing news sources to tersely report ongoing incidents, often in a single sentence. Second, since tweets are organized and presented in a stream, news sources report events as they happen rather than reporting weekly or daily summaries. Lastly, most of the sources in Figure 15 have an active and Africa-specific Twitter account[24], eliminating the need to tailor a web scraper for each source's website. Table 9 (page 42) summarizes the sources selected for ORÁCULO and the output of their scraping.

[24] We selected only the sources that had active, Africa-specific accounts. Therefore, we discarded Centrafrique Presse Info (does not have a Twitter account), aBangui and LRA Crisis Tracker (their Twitter accounts were mostly deactivated), Xinhua (does not have an Africa-specific account), and Africa 1 (contributed too little to the *ACLED event dataset*).

Table 9. Sources of the *scraped tweets dataset*. 8518 tweets were collected in total. Note the inclusion of the Office for the Coordination of Humanitarian Affairs (West and Central Africa) and the Associated Press (Africa) Twitter accounts: though they did not contribute to the *ACLED event dataset* (2019-2020), their reports generated many events in the overall ACLED CAR dataset. For this reason, their Africa-specific Twitter accounts were added to the list of selected sources to increase the amount of training/validation data.

News source	Type	Twitter account	Predominant Language	Number of scraped tweets
Radio France Internationale	International	@RFIAfrique	French	3325
Courbeau News	Local	@CorbeauNews	French	1857
Agence France-Presse	International	@AFPAfrica	English	1441
Radio Ndeke Luka	Local	@RadioNdekeLuka	French	1101
Réseau des Journalistes pour les Droits de l'Homme en Centrafrique (RJDH)	Local	@RJDH_RCA	French	288
Office for the Coordination of Humanitarian Affairs	International	@OCHAROWCA	English/French	140
Associated Press	International	@AP_Africa	English	366

Since French is one of the official languages of the Central African Republic and the *lingua franca* of most of West and Central Africa, most selected Twitter accounts post predominantly in French. We opted to translate their content to English to leverage the selected text mining resources (Table 7), and while the danger of information being quite literally "lost in translation" exists, the direct and unambiguous language and tone used in the tweets that contain news reports simplifies translation, so we consider that little loss of information due to translation is to be expected. Twitter scraping, translation, and preprocessing (removed hyperlinks, punctuation, and diacritics) were performed in a single pipeline (scraper) and its result is the *scraped tweet dataset*, which comprises 8518 tweets with the following attributes: *tweet ID*, *permalink*, *username* (tweet author), *timestamp* (tweet date), (untranslated) *content*, and *preprocessed content* (translated) (Table 10, page 49).

Finally, the *scraped tweet dataset* was manually annotated by two analysts with military and machine learning experience to create the joint *training/validation dataset*, allowing the development of the Event Extractor, and permitting a quality assessment of the Twitter data. First, scraped and translated tweets were classified according to relevance: a tweet was deemed "relevant" if it contained a conflict-related event. If that were the case, event arguments (Table 6, page 36) were then manually extracted and stored in their respective columns. Second, event codes were assigned to each tweet so that if multiple tweets reported the same event (*i.e.*, when multiple news sources report the same incident), they were assigned the same event code, allowing their aggregation into unique events. Third, tweets were classified according to their relevance to the CAR

Civil War[25]. Thus, 230 unique events related to the CAR Civil War were obtained from 8518 tweets. To visually describe the process and its results, Figure 16 shows the percentage conflict-related and CAR Civil War-related events per Twitter account, while Figure 17 shows the results of the successive phases of manual classification. As for the average daily output, the seven Twitter accounts posted an average combined daily total of 22 tweets, of which only 2 contained a relevant event (CAR and non-CAR).

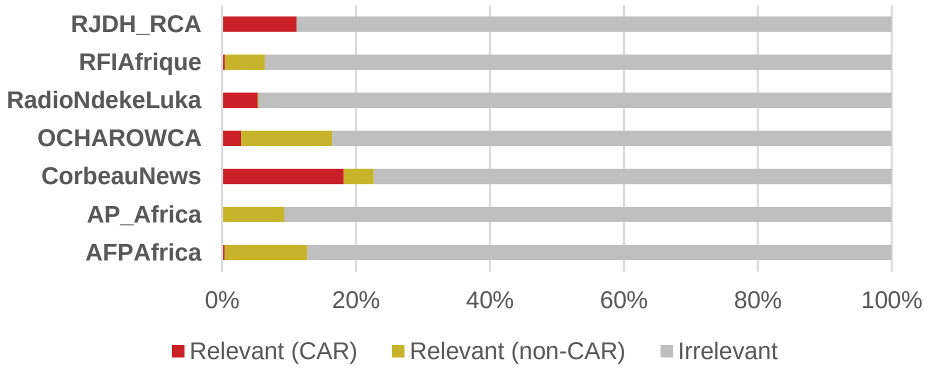


Figure 16. Percentage of conflict-related and CAR Civil War-related tweets per selected Twitter account. Note how the percentage of CAR Civil War-related tweets is much higher in the local sources.

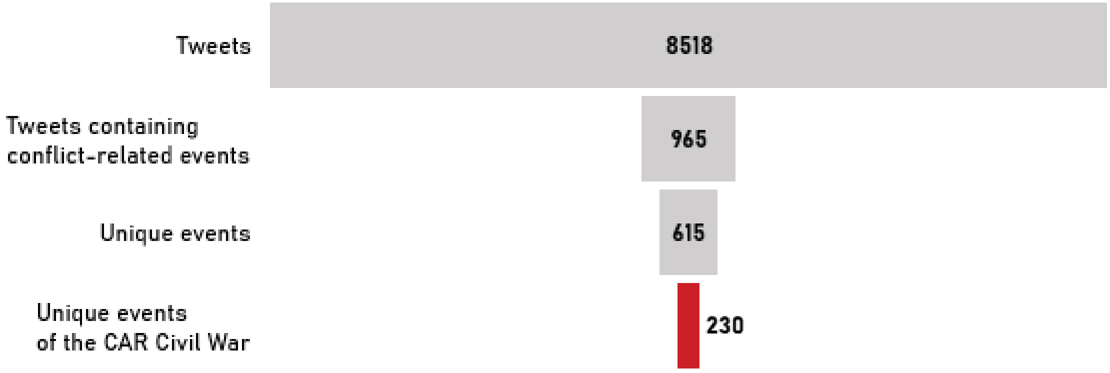
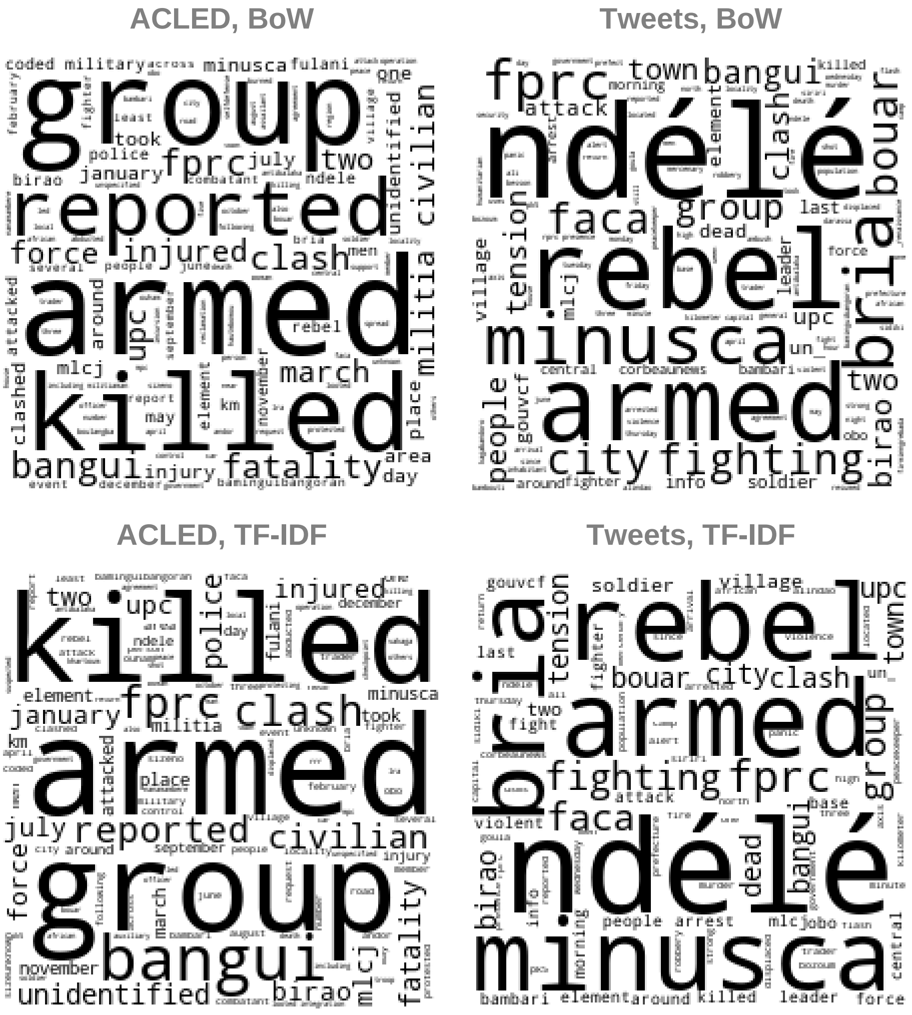


Figure 17. Results (in number of tweets/events) of the manual classification of the *scraped tweet dataset* (output: *training/validation event dataset*). Note the class imbalance between relevant and irrelevant tweets: only 11.3% of tweets (965 out of 8518) were considered as conflict-related during manual classification.

[25] Even the local sources (@CorbeauNews, @RJDH_RCA, @RadioNdekeLuka) reported events pertaining to other conflicts, like the conflicts raging in Mali, Burkina Faso, and the Democratic Republic of Congo.

4.2.3 Data Exploration and ACLED-Twitter Consistency

Once the joint _training/validation dataset_ was completed, we compared it with the _ACLED event dataset_ to assess whether the scraped data was consistent with the chosen near-ground truth (ACLED). To do so, we performed exploratory data analysis on both datasets and compared them using four dimensions: event count, content (ACLED notes/tweet content) (Figure 18), date (Figure 19, page 45), and location (Figure 20, page 45).


ACLED, BoW

Tweets, BoW

ACLED, TF-IDF

Tweets, TF-IDF

Figure 18. Word clouds of the 120 top-scoring words in the Bag of Words (BoW) and Term Frequency – Inverse Document Frequency (TF-IDF) vectorization generated from the translated tweet content of the _scraped tweets dataset_ and from the event notes (description) of the _ACLED event dataset_ (06 June 2019-16 June 2020). Each tweet/note was considered its own document, so there is little overlap between tweets and between notes vocabulary-wise. This means that the number of words with a small document frequency is high, so the "Term Frequency" component of the TF-IDF vectorization process becomes more important, and the TF-IDF word clouds greatly resembles the TF ones.

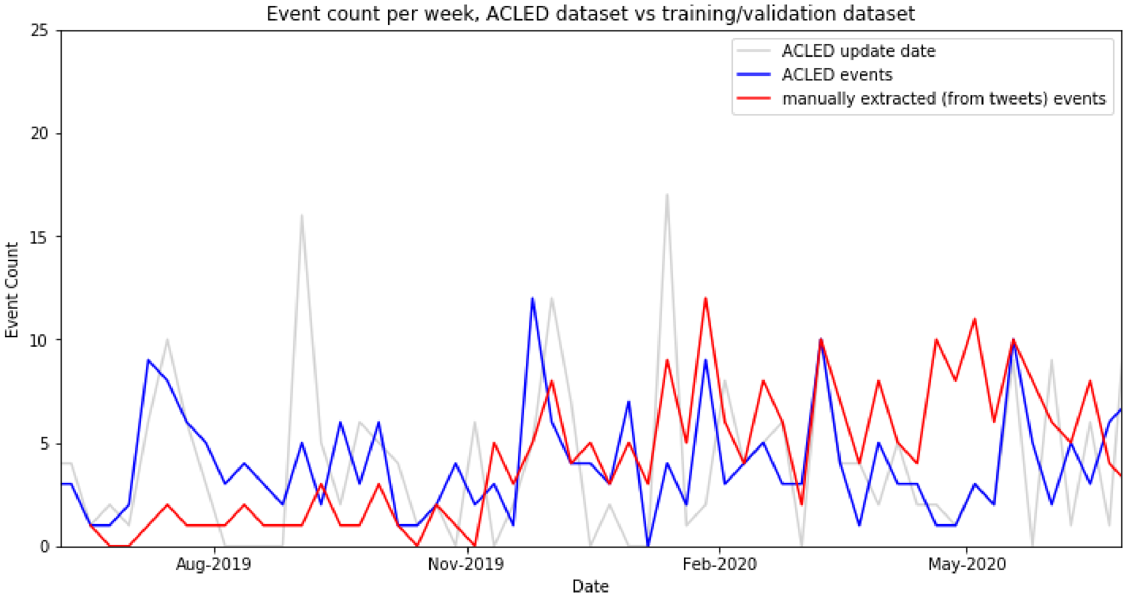


Figure 19. Graph comparing the weekly time series of events in the *ACLED event dataset* (blue) versus that of the joint *training/validation dataset* (red). The gray time series describes the date when events were *added* to the ACLED database, illustrating the lengthy ACLED reviewing process. The increase in scraped events starting in November 2019 can be attributed not only to a rise in conflict-related activity, but also to an increase in activity of the @CorbeauNews account – the most important account in terms of relevant tweets.

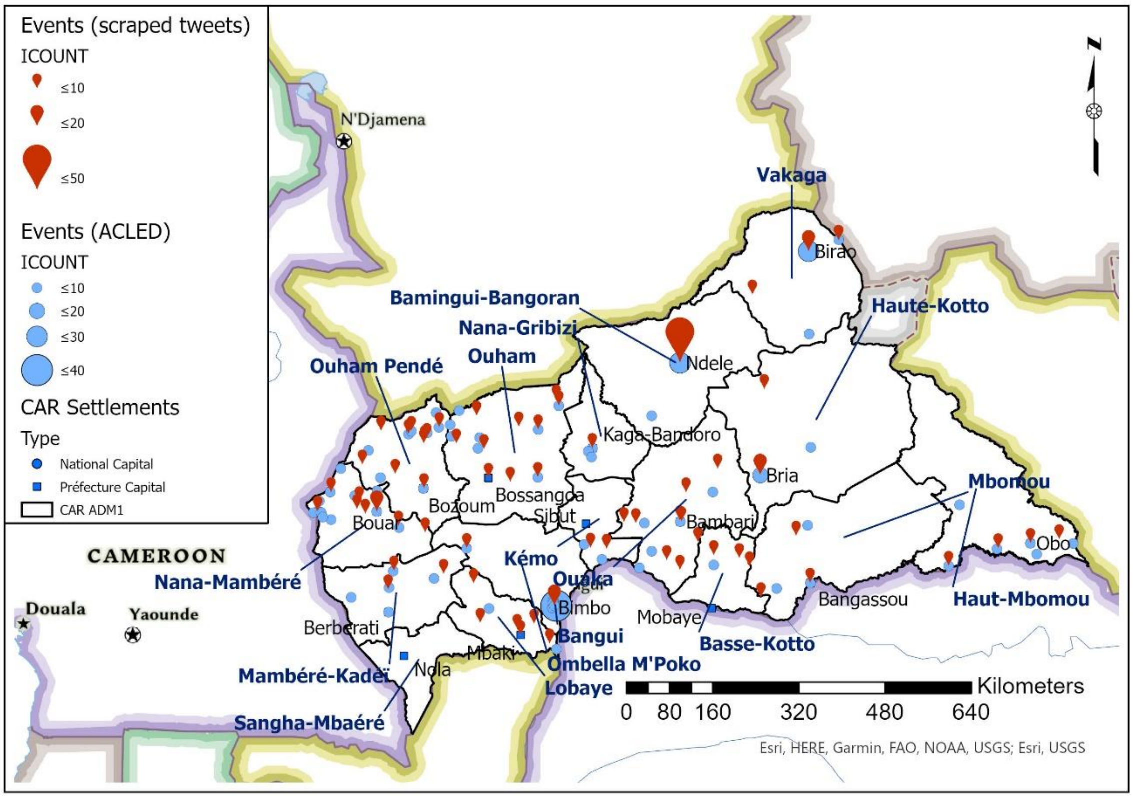


Figure 20. Map comparing event count per location in the *ACLED event dataset* (blue) and in the *training/validation dataset* (red). During the scraping interval, the city of Ndélé (capital of the Bamingui-Bangoran *préfecture*) was the stage of a high number of conflict-related events, leading to a high number of events.

The following assessments were obtained after comparing the event counts, word clouds, time series and maps of both datasets:

- Both datasets contain around the same number of events (ACLED: 222; Twitter: 230).
- Both word clouds contain the same vocabulary, albeit at different magnitudes. To illustrate, note how Bangui appears more frequently in the *ACLED event dataset* than in the *training/validation dataset*, while the opposite is true for Ndélé.
- The weekly time series of the datasets match poorly before November 2019, likely due to the inactivity of the @CorbeauNews Twitter account. After November, the peaks and valleys of the time series generally match.
- Event locations on both datasets generally match, although the *ACLED event dataset* contains more Bangui and unmatched Northwest events, while the *training/validation dataset* contains more Ndélé and unmatched South-Center events.

Therefore, while the datasets do not match perfectly, their comparison reveals that they overlap enough to validate the use of Twitter-scraped data to track the CAR Civil War and the use of ACLED data to evaluate the outputs of ORÁCULO. Lastly, the comparison allows us to outline the upper limit of the overall performance of ORÁCULO given its current input data: when using the *scraped tweet dataset* as input, a complete overlap with ACLED data is simply not possible, but a reasonable to strong match is a feasible expectation.

4.3 Event Extractor

The Event Extractor takes the *scraped tweet dataset* as input and outputs the *merged event dataset*, which comprises unique, geocoded, CAR Civil War-related events that are ready – after analyst supervision – to be stored as point objects in a geodatabase. It comprises three main functions:

- The snips NLU event extractor loops through the *scraped tweet dataset*, parsing the content of each tweet to detect events and extract their arguments. It discards the irrelevant tweets and adds the event confidence level and the event arguments as new features to the *scraped tweet dataset*, creating the *extracted event dataset*.
- The geocoder loops through the *extracted event dataset* and assigns geographic information to the extracted event locations. When multiple locations are

extracted, it decides the most relevant location. It adds the event location, level 1 administrative region (ADM1), country, latitude, and longitude (EPSG:4326) as new features to the *extracted event dataset*.

- The event merger loops through the *extracted event dataset* and merges extracted and geocoded events when they describe the same real-world event. It uses location- and date-based criteria. Its output is the *merged event dataset*, which is then inspected by an analyst to remove false positives.

4.3.1 Snips NLU Event Extractor

The snips NLU event extractor uses the Snips NLU engine to perform the tasks of detecting events in the tweets of the *scraped tweet dataset* and extracting their arguments. Despite being a NLU engine and not a dedicated event extractor, Snips NLU was selected because it is publicly available, adaptable, and hybrid: when compared to the state-of-the-art event extraction systems reviewed in Chapter 3, only Snips NLU and Giveme5W1H made their code public, but of the two systems, Snips NLU allows an easier definition of custom entities (recall that we are using a custom "conflict-related event" frame with custom arguments) and uses both pattern matching and machine learning to extract intents/events. Thus, a Snips NLU-based event extractor should leverage low amounts of training data (the annotated scraped tweets) and preexisting expert knowledge (factions, toponyms, event types) better than other state-of-the-art solutions.

In our implementation of the Snips NLU engine – the snips NLU event extractor function – we use the "English" configurations to leverage the pretrained word embeddings and define and populate four custom[26] entities: *action_word* for the *Action* argument (populated with NATO Mission Task Verbs[27] and related words), *faction_word* for *Agent* and *Target* (populated with known CAR Civil War faction names), *location_word* for *Location* (populated with CAR settlement names), and *effect_word* for *Effects*. Then, after defining the custom entities, utterances are populated with annotated relevant tweets taken from the *training/validation dataset*.

[26] Though Snips NLU offered the built-in *snips.city*, *snips.country* and *snips.region* entities, the combined nature of the *Location* argument (villages, cities, regions, and countries) and the possibility of discarding poorly recorded Central African towns and villages led us to define a custom *location_word* entity. Furthermore, though the *snips.datetime* entity was defined, and though temporal references existed in the scraped tweets (*i.e.*, "two days ago"), during the creation of the *training/validation* dataset we found that the tweet dates themselves corresponded overwhelmingly to the true event dates, so they were directly recorded as such for simplicity.

[27] These verbs describe common actions performed by combatants in an armed conflict, such as "attack" or "ambush".

To combat the low amount of training data, we consider every conflict-related tweet in the *training/validation dataset* as a training utterance, as though the *Agent*, *Target* and *Location* arguments of CAR Civil War and non-CAR Civil War tweets differ, their underlying structure (*i.e.*, the type of sentence and the words around the arguments) is mostly the same, expanding the effective training set. Finally, entities and utterances are combined into a single training file and fed to the Snips NLU engine. Figure 21 illustrates the training procedure.

```
---
type: entity
  name: action_word
  automatically_extensible: true # default value is true
  use_synonyms: true # default value is true
  matching_strictness: 1.0 # default value is 1.0
values:
   - arrest
   - fighting
   - battle
   - explosion
   - protest
```

```
---
type: intent
name: event
  slots:
   - name: action
    entity: action_word
   - name: agent
    entity: faction_word
   - name: target
    entity: faction_word
   - name: date
    entity: snips/datetime
   - name: location
    entity: location_word
   - name: effect
    entity: effect_word

utterances:
   - Central African Republic [action](Fighting resumed) again [date](this
morning) in [location](Obo) in Haut-Mbomou between the [target](FACA) and
[agent](Ali Darasss UPC). The fighting is approaching dangerously close to the
city and Obo would risk falling into the hands of the rebel fighters.
   - RCA [action](fighting resumes) in [location](Obo) the [agent](rebels) are in
the city and the [target](Faca) are retreating.
```

Figure 21. Excerpts of the YAML training file used by the snips NLU event extractor. The first excerpt illustrates how to define entities and how to populate them with a set of initial values (gazetteer). The second excerpt illustrates how to define intents using the entities defined above and the annotated utterances (the training set). Note how the utterances do not have to use the entity values define above.

Once the Snips NLU engine is trained, it is ready to parse tweet content. For each tweet in the input dataset, Snips NLU parses its content and classifies each n-gram as either an event argument or as an uninformative feature (each argument can contain multiple n-grams). It then assigns an event confidence level to the tweet, which is used by the snips NLU event extractor wrapper function to decide whether the tweet contains a conflict-related event: if the confidence level is above a heuristically determined threshold, the tweet is considered as relevant and preserved; otherwise, it is discarded. The output of the entire process is the input tweet dataset, but without irrelevant tweets, and with the event arguments and event confidence level added as new features (Table 10).

Table 10. Features of the *scraped tweet dataset* and *extracted event dataset*. The functions of the Event Extractor add the features describes below to the *scraped tweets dataset*, generating the *extracted event dataset*. Also note that the *extracted event dataset* does not contain irrelevant tweets.

Feature (alias)	Description	Origin
tweet id	ID assigned by Twitter to each tweet.	scraper
username	Tweet author.	
permalink	Link to the tweet.	
timestamp	Timestamp when the tweet was posted.	
content	Original tweet text.	
preprocessed content	Translated and preprocessed (no punctuation, hyperlinks, or diacritics) tweet text.	
event confidence level (conf_lvl)	0.000 to 1.000. Considered relevant if above the chosen threshold *(see 5.1.1 for further discussion)*.	snips NLU event extractor
action (pred_action)	List of extracted n-grams for the *Action* argument.	
agent (pred_agent)	List of extracted n-grams for the *Agent* argument.	
target (pred_target)	List of extracted n-grams for the *Target* argument.	
date	List of extracted n-grams for the *Date* argument. Equal to the tweet *timestamp*.	
location (pred_loc)	List of extracted n-grams for the *Location* argument.	
effects (pred_fx)	List of extracted n-grams for the *Effects* argument.	
event location	Most relevant of the values stored in the *location* list.	geocoder
adm1	Level 1 administrative region of the *event location*.	
Country	Country of the event location.	
type code	*event location* type (0: country to 3: town)	
latitude	Latitude (EPSG:4326) of the *event location*.	
longitude	Longitude (EPSG:4326) of the *event location*.	
event code	ID of the unique event.	event merger

4.3.2 Geocoder

For each event in the *extracted event dataset*, the geocoder loops through the list of candidate extracted locations, selects the one that best describes the place where the event took place – the *event location* – and completes its geographic information. This is performed in a two-stage process:

- First, the loc finder function attempts to complete the geographic information (*ADM1*, *country*, *type code*, *latitude*, and *longitude*) of every candidate location stored in the *location* feature of the *extracted event dataset*.
- Then, the loc selector function decides which of the candidate extracted locations best describes the true *event location*, storing its geographic information as new features in the *extracted event dataset*.

To geocode candidate locations, the loc finder function combines the GeoNames[28] geocoding web service and a CAR gazetteer created from the UN OCHA CAR settlements dataset (Figure 22, page 51). After preprocessing the extracted locations[29], the first and second stages of loc finder are calls to the GeoNames web service: the first is the strictest orthography-wise and targets countries exclusively (*i.e.*, a city such as "Bangui" will not be detected); the second is slightly less strict and targets cities and administrative regions in Africa, with a bias to the CAR (*i.e.*, between two similarly named places, loc finder will select the one in the CAR first). Then, if the extracted location is not geocoded, the third stage searches in the CAR gazetteer. Since there is no standard orthography for many Central African toponyms (Table 11, page 51), the third stage does not simply search for a matching toponym in the CAR gazetteer; rather, it sorts the known toponyms by ascending edit distance[30] from the extracted location, returning those that are less than three edits away *and* that still match its double metaphone[31] pattern (this is to ensure that the edits do not result in toponyms with a completely different sound). Finally, if nothing is found in the gazetteer, the extracted location is presented to GeoNames one

[28] Several geocoding web services were tested, but GeoNames proved the best at recognizing Central African place names, leading to its selection as the primary geocoding service.

[29] Preprocessing involves removing any unrelated, lowercase words that were unwittingly extracted (for instance, if the snips NLU event extractor extracted "the national capital Bangui" as the *location* argument). It also converts some common urban districts (*i.e.*, Bangui's PK5 district) to the name of the city in which they are located.

[30] The edit distance (Левенштейн, 1965) between two strings is the minimum number of operations (*i.e.*, add/change/remove characters) needed to turn one string into the other.

[31] The double metaphone algorithm represents words according to their approximate pronunciation, allowing us to retrieve an event location from the gazetteer even if the spelling of the extracted location is radically different, but pronounced similarly.

last time, but matching strictness is lax (although the bias is still the CAR), and results are limited to a bounding box that covers West and Central Africa. If no known place is found by the end of the fourth stage, loc finder returns "None".

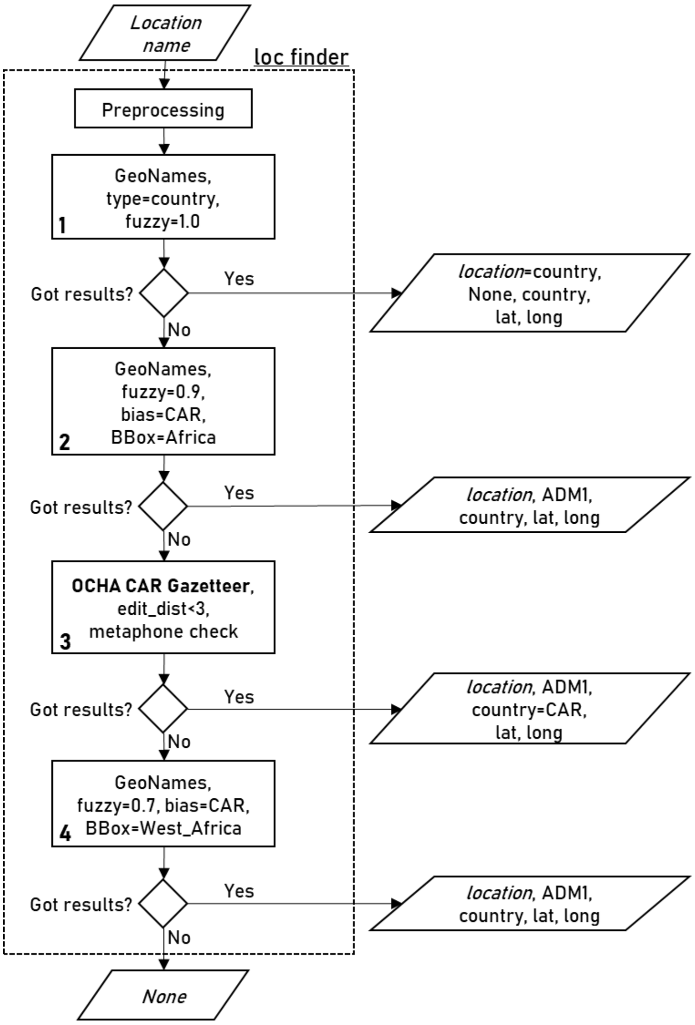
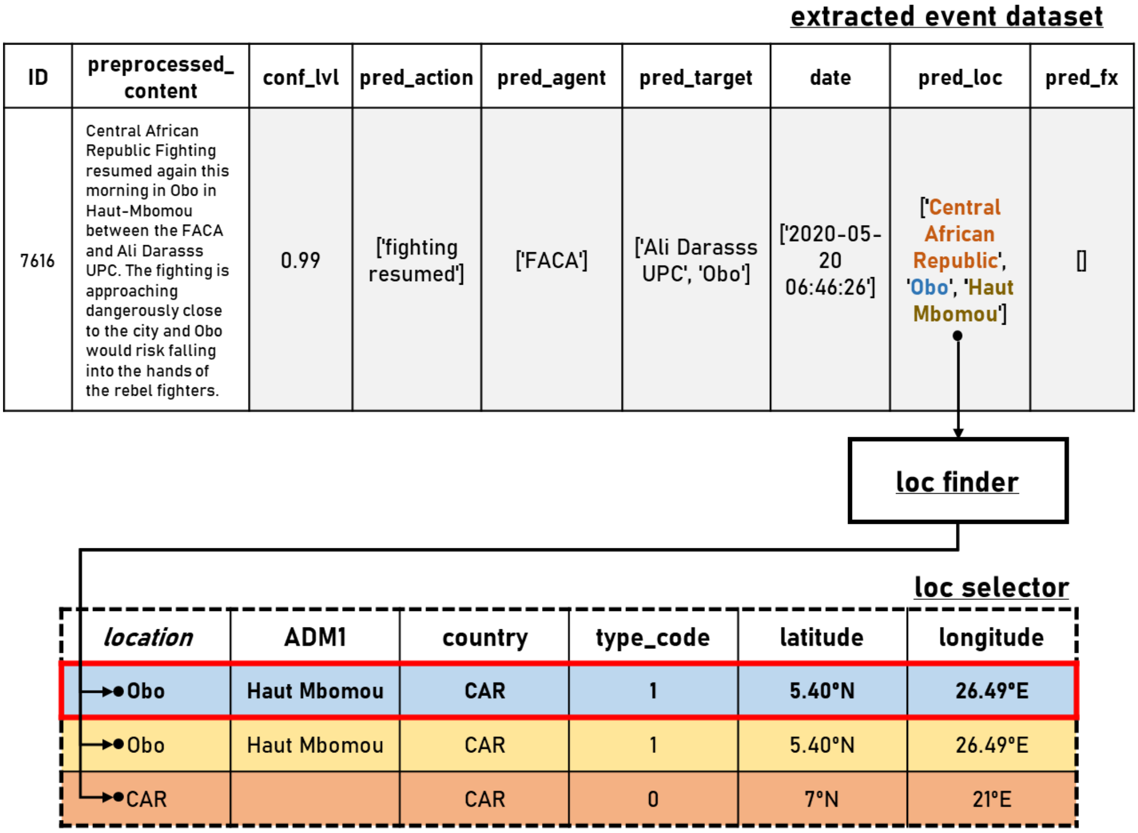

Location name
loc finder
Preprocessing
GeoNames, type=country, fuzzy=1.0
1
Got results?
Yes
No
location=country, None, country, lat, long
GeoNames, fuzzy=0.9, bias=CAR, BBox=Africa
2
Got results?
Yes
No
location, ADM1, country, lat, long
OCHA CAR Gazetteer, edit_dist<3, metaphone check
3
Got results?
Yes
No
location, ADM1, country=CAR, lat, long
GeoNames, fuzzy=0.7, bias=CAR, BBox=West_Africa
4
Got results?
Yes
No
location, ADM1, country, lat, long
None

Figure 22. Diagram of the geocoder.loc finder function (ADM1 refers to the level 1 administrative region; BBox refers to the bounding box). The "fuzzy" parameter in the GeoNames web service controls the matching strictness: the higher the parameter, the stricter will the geocoder enforce the orthography of the input location name. Note how the matching strictness gets progressively laxer in subsequent stages (geocoding attempts).

Table 11. Example of the difficulties faced when geocoding Central African place names.

Extracted Location	Geocoding system	Geocoding result	Error	Solution	Event Location
Gbokologbo	GeoNames web service	None	Place name not found	Try CAR gazetteer	Bokolobo, Ouaka *préfecture*, CAR, 5.42N 20.95E
	CAR gazetteer	None	Transliteration (silent "G")	Try again without silent "G"	

After geocoding each candidate location, the loc selector function (Figure 23) uses a heuristic to select the one that best describes the related event: the most specific (*i.e.*, the smallest) out of the candidate locations is usually the *event location*, while all the others provide points of reference. Consequently, loc selector creates a decision table from the candidate locations, selects those whose parent ADM1 and country are the most likely (*i.e.*, the most common), and sorts the result according to location type code[32]: the highest the type code, the smallest the location. Then, the location whose type code is highest is selected as the *event location*, and its name, parent ADM1, country, type code, and geographic coordinates (EPSG:4326) are added to *extracted event dataset* as new features.



ID	preprocessed_content	conf_lvl	pred_action	pred_agent	pred_target	date	pred_loc	pred_fx
7616	Central African Republic Fighting resumed again this morning in Obo in Haut-Mbomou between the FACA and Ali Darass UPC. The fighting is approaching dangerously close to the city and Obo would risk falling into the hands of the rebel fighters.	0.99	['fighting resumed']	['FACA']	['Ali Darasss UPC', 'Obo']	['2020-05-20 06:46:26']	['Central African Republic', 'Obo', 'Haut Mbomou']	[]

location	ADM1	country	type_code	latitude	longitude
Obo	Haut Mbomou	CAR	1	5.40°N	26.49°E
Obo	Haut Mbomou	CAR	1	5.40°N	26.49°E
CAR		CAR	0	7°N	21°E

Figure 23. Diagram of the geocoder.loc selector function. Note how the geocoder.loc finder function geocodes “Haut Mbomou” (a *préfecture*) as its capital, Obo. As the type code of Obo is the highest, it was selected as the final event location.

[32] Types were assigned to locations according to the following system adopted from the OCHA CAR Gazetteer: countries – 0; *préfectures* and their capitals – 1; *sous-préfectures* and their capitals – 2; *communes*, their capitals, and smaller settlements – 3.

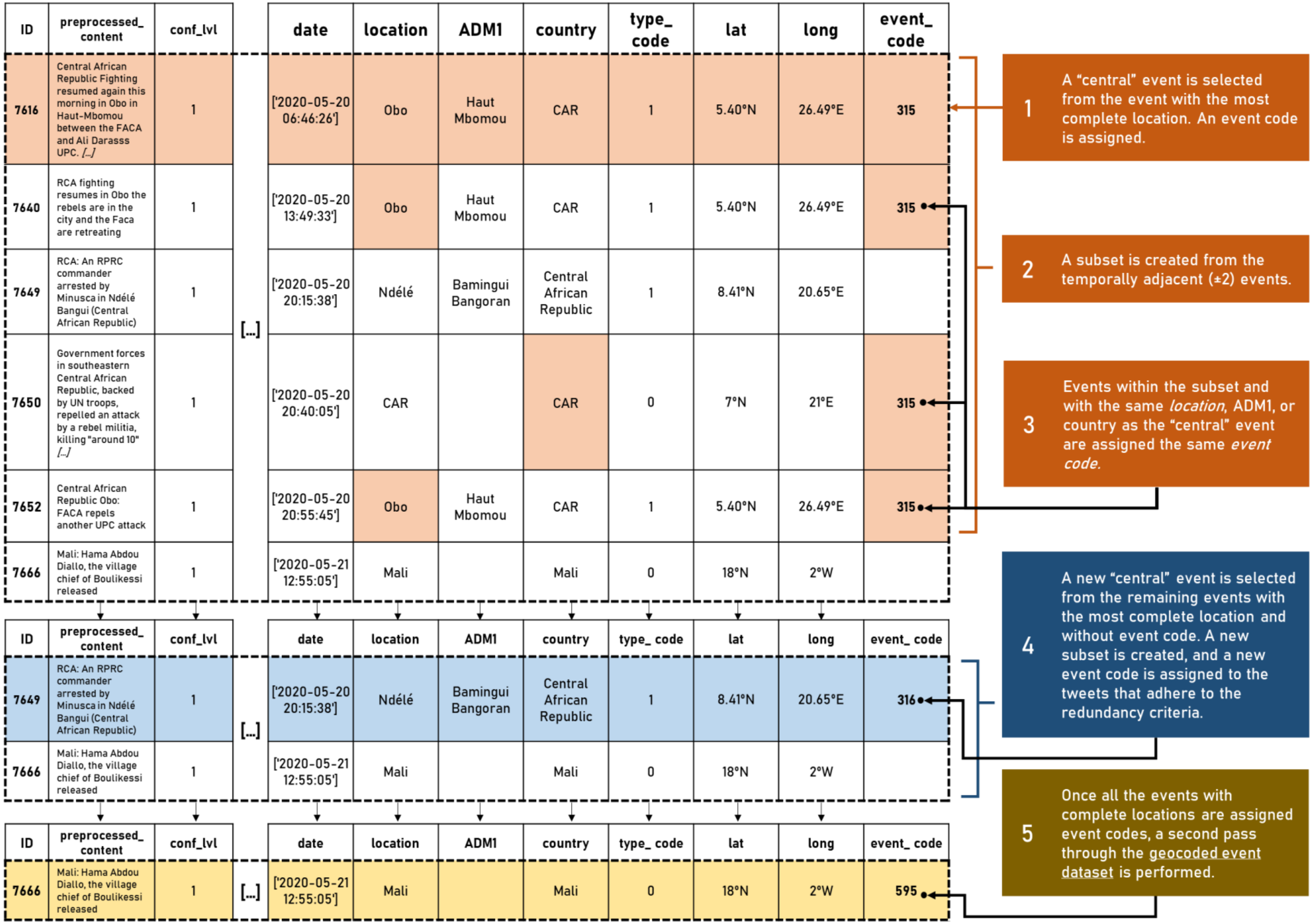
4.3.3 Event Merger

The event merger merges the redundant extracted and geocoded events of the *extracted event dataset* into unique events of the *merged event dataset*. As with the geocoder.loc selector function, event merger employs a heuristic: rather than merging tweets into events by comparing every argument (which would require comparing subjective arguments such as *action*), it assumes that if two different extracted events describe events that happened in the same *event location* and around the same *date* (±1 day), they are likely reporting the same real-world event.

Thus, event merger starts by selecting the events with the most complete geographic information (*i.e.*, events with a town/city-level *event location* and a known *ADM1*). Then, for each of those events (the "central" event), it assigns an *event code* and selects a subset of the entire *extracted event dataset* comprising every event whose *date* falls within a 1-day interval from the "central" event. Next, it assigns the same *event code* as the "central" event to the events of that subset that share with it their *event location* or *ADM1*, or whose *event location* is the name of the *ADM1* or *country* of the "central" event. As this "first pass" loops only through the most complete events, it is likely that afterwards many events will still lack an *event code*, as they are either not temporally adjacent to the most specific events or have failed the merging criteria. Thus, a second pass through the dataset is performed, looping through every remaining event that lacks an *event code*. Figure 24 (page 54) illustrates the how the assignment of event codes works.

Finally, once the event merger assigns an *event code* to every event of the *extracted event dataset*, events are grouped by *event code* to form the *merged event dataset*, the output of the Event Extractor. During this merging process, when multiple extracted events with the same *event code* are merged into a single merged event, their arguments are stored in lists. Duplicate values are removed from these argument lists, but they can still contain several different *dates* and even several different *event locations*. To solve the problem of multiple event *dates*, only the earliest is preserved[33]. Likewise, to solve the problem of multiple *event locations*, the heuristic of the geocoder.loc selector function is employed again: the location whose type code is highest are selected as the final *event location* and stored in the *merged event dataset*.

[33] The earliest mention of an event is the closest to the actual event date, as no news source can report an event *before* it happens.



ID	preprocessed_ content	conf_lvl		date	location	ADM1	country	type_ code	lat	long	event_ code
7616	Central African Republic Fighting resumed again this morning in Obo in Haut-Mbomou between the FACA and Ali Darasss UPC. *[...]*	1	[...]	['2020-05-20 06:46:26']	Obo	Haut Mbomou	CAR	1	5.40°N	26.49°E	315
7640	RCA fighting resumes in Obo the rebels are in the city and the Faca are retreating	1		['2020-05-20 13:49:33']	Obo	Haut Mbomou	CAR	1	5.40°N	26.49°E	315
7649	RCA: An RPRC commander arrested by Minusca in Ndélé Bangui (Central African Republic)	1		['2020-05-20 20:15:38']	Ndélé	Bamingui Bangoran	Central African Republic	1	8.41°N	20.65°E	
7650	Government forces in southeastern Central African Republic, backed by UN troops, repelled an attack by a rebel militia, killing "around 10" *[...]*	1		['2020-05-20 20:40:05']	CAR		CAR	0	7°N	21°E	315
7652	Central African Republic Obo: FACA repels another UPC attack	1		['2020-05-20 20:55:45']	Obo	Haut Mbomou	CAR	1	5.40°N	26.49°E	315
7666	Mali: Hama Abdou Diallo, the village chief of Boulikessi released	1		['2020-05-21 12:55:05']	Mali		Mali	0	18°N	2°W	

1 A "central" event is selected from the event with the most complete location. An event code is assigned.

2 A subset is created from the temporally adjacent (±2) events.

3 Events within the subset and with the same *location*, ADM1, or country as the "central" event are assigned the same *event code*.

ID	preprocessed_ content	conf_lvl		date	location	ADM1	country	type_ code	lat	long	event_ code
7649	RCA: An RPRC commander arrested by Minusca in Ndélé Bangui (Central African Republic)	1	[...]	['2020-05-20 20:15:38']	Ndélé	Bamingui Bangoran	Central African Republic	1	8.41°N	20.65°E	316
7666	Mali: Hama Abdou Diallo, the village chief of Boulikessi released	1		['2020-05-21 12:55:05']	Mali		Mali	0	18°N	2°W	

4 A new "central" event is selected from the remaining events with the most complete location and without event code. A new subset is created, and a new event code is assigned to the tweets that adhere to the redundancy criteria.

ID	preprocessed_ content	conf_lvl		date	location	ADM1	country	type_ code	lat	long	event_ code
7666	Mali: Hama Abdou Diallo, the village chief of Boulikessi released	1	[...]	['2020-05-21 12:55:05']	Mali		Mali	0	18°N	2°W	595

5 Once all the events with complete locations are assigned event codes, a second pass through the geocoded event dataset is performed.

Figure 24. Diagram of the event merger function.

4.4 Event Dataset Supervision

Before being fed to the Hot Spots Detector, event data in the *merged event dataset* is supervised by analysts. Their primary tasks are discarding false positives (irrelevant events) and correcting extracted arguments, creating the *refined event dataset*. Their secondary task is to annotate the content of correctly extracted events, allowing their inclusion in the snips NLU event extractor training file. Therefore, analyst supervision directly increases the spatial, temporal, and thematic precisions of the Hot Spots Detector by controlling the quality of its input data, but also indirectly increases them by improving the training of the Event Extractor.

4.5 Hot Spots Detector

The Hot Spots Detector component analyses the events extracted from online news sources by the Event Extractor, detecting spatiotemporal hot spots of events and their temporal trends. Its input is the *refined event dataset*, and its outputs are the *event geodatabase* and the *event space-time-cube*. It comprises two functions:

- The geodatabase generator stores the events of the *refined event dataset* as point features in the *event geodatabase*.
- The emerging hot spots detector aggregates the events in the *event geodatabase* into the *event space-time cube* and uses the Getis-Ord Gi* statistic (*e.g.*, ESRI, 2020a) and the Mann-Kendall Trend test (*e.g.*, HydroGeoLogic, Inc., 2005) to discover spatiotemporal hot spots of events and their temporal trends. Its output is the *event space-time cube* with the test results added as new features.

4.5.1 Geodatabase Generator

The geodatabase generator function contains the set of ArcPy scripts necessary to generate, populate and update the *event geodatabase*. To populate the *event geodatabase*, the CAR events of the *refined event dataset* are loaded as point features using their geographic coordinates (non-CAR events are not loaded). The resulting point feature class is then projected from the WGS84 GCS (EPSG:4326) to the WGS 84/UTM zone 34N Projected Coordinate System (PCS) (EPSG:32634), ensuring consistency with the contextual information in the *event geodatabase*: the *CAR settlements* point feature class and the *CAR ADM1*, *CAR ADM2*, and *CAR ADM3* polygon feature classes. Finally, the *event geodatabase* also stores the 2-dimensional and 3-dimensional visualizations of *event space-time cube* after being subjected to emerging hot spots analysis (4.5.2) –

in short, every spatial output of ORÁCULO save for the space-time cubes themselves is stored in the event geodatabase, whose schema is illustrated below (Figure 25).

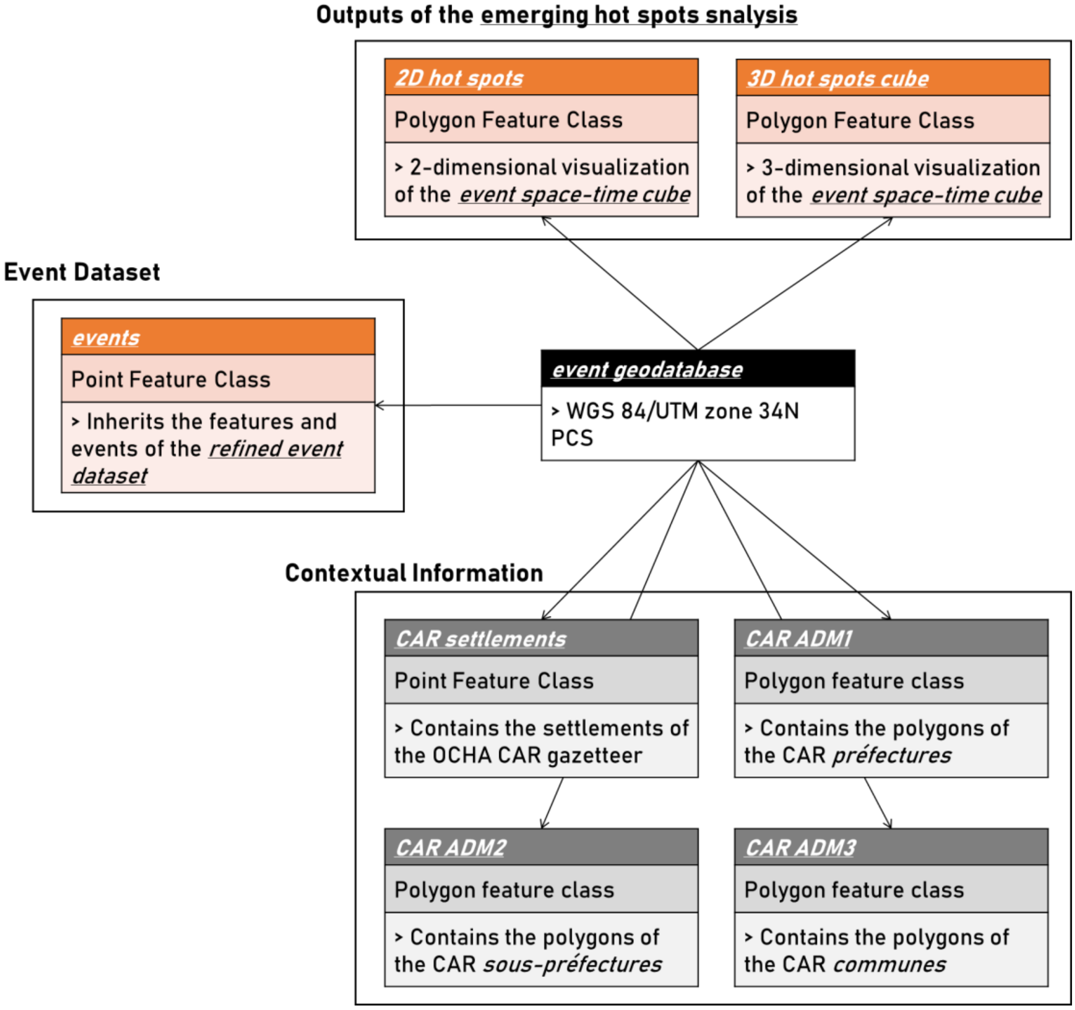

Outputs of the emerging hot spots snalysis
2D hot spots
Polygon Feature Class
> 2-dimensional visualization of the event space-time cube
3D hot spots cube
Polygon Feature Class
> 3-dimensional visualization of the event space-time cube
Event Dataset
events
Point Feature Class
> Inherits the features and events of the refined event dataset
event geodatabase
> WGS 84/UTM zone 34N PCS
Contextual Information
CAR settlements
Point Feature Class
> Contains the settlements of the OCHA CAR gazetteer
CAR ADM1
Polygon feature class
> Contains the polygons of the CAR préfectures
CAR ADM2
Polygon feature class
> Contains the polygons of the CAR sous-préfectures
CAR ADM3
Polygon feature class
> Contains the polygons of the CAR communes

Figure 25. Schema of the *event geodatabase*.

4.5.2 Emerging Hot Spots Detector

The emerging hot spots detector function detects statistically significant spatiotemporal event hot spots and their temporal trends. To do so, it performs a two-stage process:

- First, it uses the “Create Space-Time Cube By Aggregating Points” ArcGIS Pro geoprocessing tool to aggregate the *events* point feature class of the *event geodatabase* into bins of the *event space-time cube* (Figure 26).
- Then, it deploys the “Emerging Hot Spots Analysis” ArcGIS Pro geoprocessing tool to compute the Getis-Ord Gi* statistic **(1)** (page 55) of each cube bin from the bin event count and the Mann-Kendall Trend Test statistic **(2)** (page 56) of

each cube column (Figure 27) from the Gi* of each cube bin, adding the resulting hot spot and trend z-scores as new features to the *event space-time cube* (Table 12, page 58). Bins with a high Gi* value are considered part of an event hot spot, and their temporal trend (up to the most recent time step) is given by the Mann-Kendall statistic.

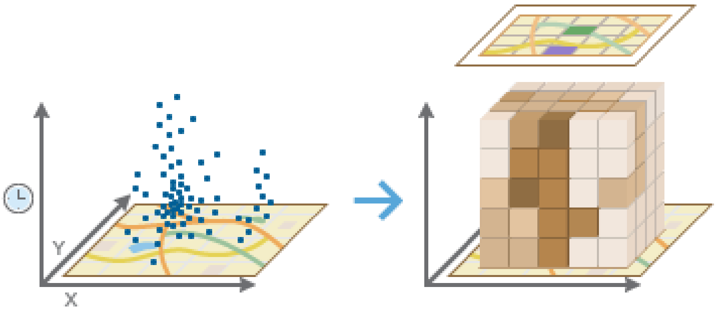


Figure 26. Diagram illustrating how timestamped point events are aggregated into the bins of a space-time cube. *Source:* (ESRI, 2020b).

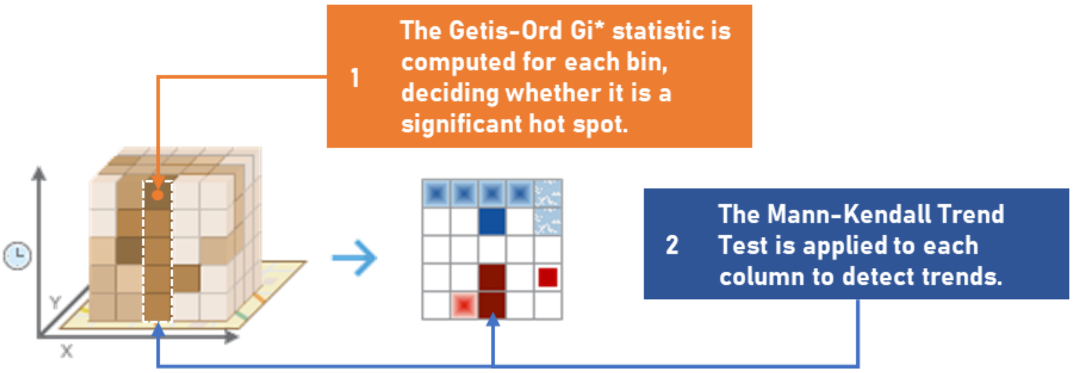

1
The Getis-Ord Gi* statistic is computed for each bin, deciding whether it is a significant hot spot.
2
The Mann-Kendall Trend Test is applied to each column to detect trends.
Y
X

Figure 27. Diagram explaining how the emerging hot spots analysis works. *Source:* Adapted from (ESRI, 2020e).

$$G_I^* = \frac{\sum_{j=1}^{n} w_{i,j}x_j - \bar{X}\sum_{j=1}^{n} w_{i,j}}{S\sqrt{\frac{n\sum_{j=1}^{n} w_{i,j}^2 - \left(\sum_{j=1}^{n} w_{i,j}\right)^2}{n-1}}} \qquad \textbf{(1)}$$

The Getis-Ord Gi* statistic (z-score) of feature *i* (bin) with an attribute value of x_i (event count) is given by equation **(1),** where:

$\bar{X}$ represents the population mean of the attribute being measured (event counts);
S represents the population standard deviation of the attribute; *n* represents the size of the population;
$w_{i,j}$ represents the weight between features *i* and *j*.

In the emerging hot spots analysis, the weight between feature *i* and the features of its neighborhood is 1, while the weight between *i* and all other features is 0.
Sources: (ESRI, 2020a)

$$MK_Z = \begin{cases} \dfrac{MK - 1}{\sqrt{var(S)}} if\ S > 0 \\ 0\ if\ S = 0 \\ \dfrac{MK - (-1)}{\sqrt{var(S)}} if\ S < 0 \end{cases} \tag{2}$$

$$MK = \sum_{k=1}^{n-1} \sum_{j=k+1}^{n} sign(x_j - x_k) \tag{2.1}$$

$$sign(x_j - x_k) = \begin{cases} 1\ if\ x_j - x_k > 0 \\ 0\ if\ x_j - x_k = 0 \\ -1\ if\ x_j - x_k < 0 \end{cases} \tag{2.2}$$

The result (z-score, variable MK_z) of the Mann-Kendall Trend Test of a time series (cube column) which comprises features k to n (column bins) with attribute values between x_k and x_n (Gi* of column bins) is given by equations **(2), (2.1)** and **(2.2)**, where:

var(MK) represents the variance of the Mann-Kendall test statistic, *S*.

Source: (HydroGeoLogic, Inc., 2005).

Table 12. Selected features of the *event space-time cube* and *ACLED event space-time cube*.

Feature	Description	Origin
location	ID of each grid square.	space-time cube generator
time step	ID of each time step.	
event count	Number of events that happened in the *location* during the *time step*.	
emerging hot spots count z-score	Result of the Getis-Ord Gi* test for the bin, computed using the *event count*.	emerging hot spots analysis
emerging hot spots count p-value	p-value of the *emerging hot spots count z-score*.	
trend count z-score	Result of the Mann-Kendall test for the column, computed using the *emerging hot spots count z-score*.	
trend count p-value	p-value of the *trend count z-score*.	

The main parameters of the emerging hot spots detector ("cube parameters") are bin shape, spatial and temporal bin sizes, and spatial and temporal neighborhood sizes. In the context of ORÁCULO, a space-time cube bin is a homogenous region of space (bin shape and spatial bin size) and time (temporal bin size) regarding conflict-related activity, and its neighborhood is the set of bins that fall within a certain relevant distance (spatial neighborhood size) from it, and that either happened during its time step, or up to a certain relevant number of time steps before it (temporal neighborhood size). From equation **(1)**, we can see how they affect the Gi* of each bin:

- Bin shape (square or hexagonal grid) and bin size (spatial and temporal) affect event count per bin, which is the attribute value (x_i) of each bin.
- Neighborhood size (spatial and temporal) affects how many spatial and temporal neighbors of any given bin will receive a non-null weight ($w_{i,j}$).

The best cube parameters are found after performing parameter sweep across a set of possible cube parameters. Possible cubes parameters include default parameters provided by the "Create Space-Time Cube By Aggregating Points" and "Emerging Hot Spot Analysis" tools and parameters chosen after computing the spatial and temporal autocorrelations[34] of the input datasets. The performance metrics used in the parameter sweep are the Pearson Correlation Coefficients of the *emerging hot spots count z-score* and *trend count z-score* between an *event space-time cube* and an *ACLED event space-time cube* generated using the same test cube parameters. The reasoning for this choice of metric is that since both the events coded by ACLED and the events extracted by ORÁCULO during the scraping period can be regarded as "samples" of the "population" of CAR Civil War events, if we assume that the distributions of the ACLED and ORÁCULO samples are sufficiently like the distribution of the population *and* that each constitutes a large enough sample, their event hot spots (bin Gi* value) and trends (column Mann-Kendall value) should be similar and similar to those of the population, so a high correlation between "sample" event space-time cubes should translate to a high correlation with the hypothetical "population" event space-time cube, *i.e.*, the "true" event hot spots. Below, Figure 28 illustrates the reasoning behind the adoption of the ACLED-ORÁCULO correlation as the parameter sweep performance metric.

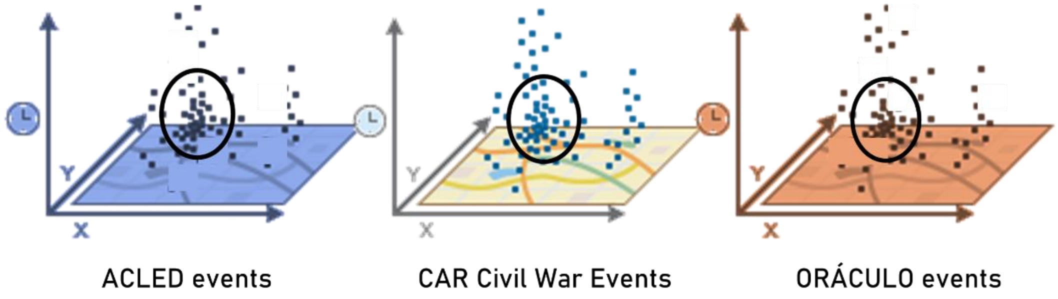


Figure 28. Diagram explaining the ACLED-ORÁCULO correlation performance metric. While ACLED and ORÁCULO events are dissimilar samples of the CAR Civil War events population, if their distributions are similar enough, event hot spots and trends should be similar as well. *Source:* Adapted from (ESRI, 2020e).

[34] While several methods to evaluate the spatiotemporal autocorrelation of a dataset have been proposed (Yong, Jing, Haohan, & Yu, 2019), none were mature enough to be included in the ArcGIS Pro geoprocessing toolset that was used to analyze spatial data; thus, spatial, and temporal autocorrelations were evaluated separately.

CHAPTER OUTPUT

In chapter 4, we described how the ORÁCULO prototype implements the state-of-the-art event extraction and spatiotemporal analysis methods reviewed in chapter 3 to address the limitations to the situational awareness of MINUSCA identified in chapter 2. Additionally, we reviewed the data used to develop and test the prototype. In the next chapter, we describe the test process and its results, and discuss whether the prototype compares favorably with the state-of-the-art reviewed in chapter 3 and whether it addresses the constraints identified in chapter 2.

5 Results and Discussion

Chapter Goals

The goals of Chapter 5 are describing the tests conducted to evaluate ORÁCULO and assessing their results given the state-of-the-art described in Chapter 3 and the context described in Chapter 2. It maps to the "Generate Test Design" and "Assess Model" tasks of the "Modelling" phase of CRISP-DM and to the "Evaluate Results" task of its "Evaluation" phase.

5.1 Results

The performance of ORÁCULO was assessed by testing its components and key functions against manually classified (*training/validation dataset*) or near-ground truth (ACLED) datasets. The following tests were performed:

- To assess the snips NLU event extractor, the manually annotated *training/validation dataset* was used to perform cross-validation. Performance was optimized by testing three strategies and tuning two parameters, and the two best configurations were compared on event detection performance with Logistic Regression and SVM classifiers trained and tested on the same *training/validation dataset* using the same cross-validation procedure.
- To assess the geocoder, we fed it the *training/validation dataset* and compared the geographic information it assigned with the manually assigned geographic information.
- To assess the event merger, we fed it the *training/validation dataset* and compared its output with that of the manual event merging.
- To assess the potential of ORÁCULO for event extraction from online news sources, we computed the overlap between the *refined event dataset* (CAR events only) and the *ACLED event dataset*.
- To assess the potential of ORÁCULO for detecting hot spots of events extracted from online news sources, we computed the correlation of hot spots and trend z-scores between the *event space-time cube* and *ACLED space-time cube* across a range of possible cube parameters. Possible parameters were obtained by the default ArcGIS methods and by assessing the temporal and spatial autocorrelations of the CAR *event geodatabase* and the *ACLED event geodatabase*.

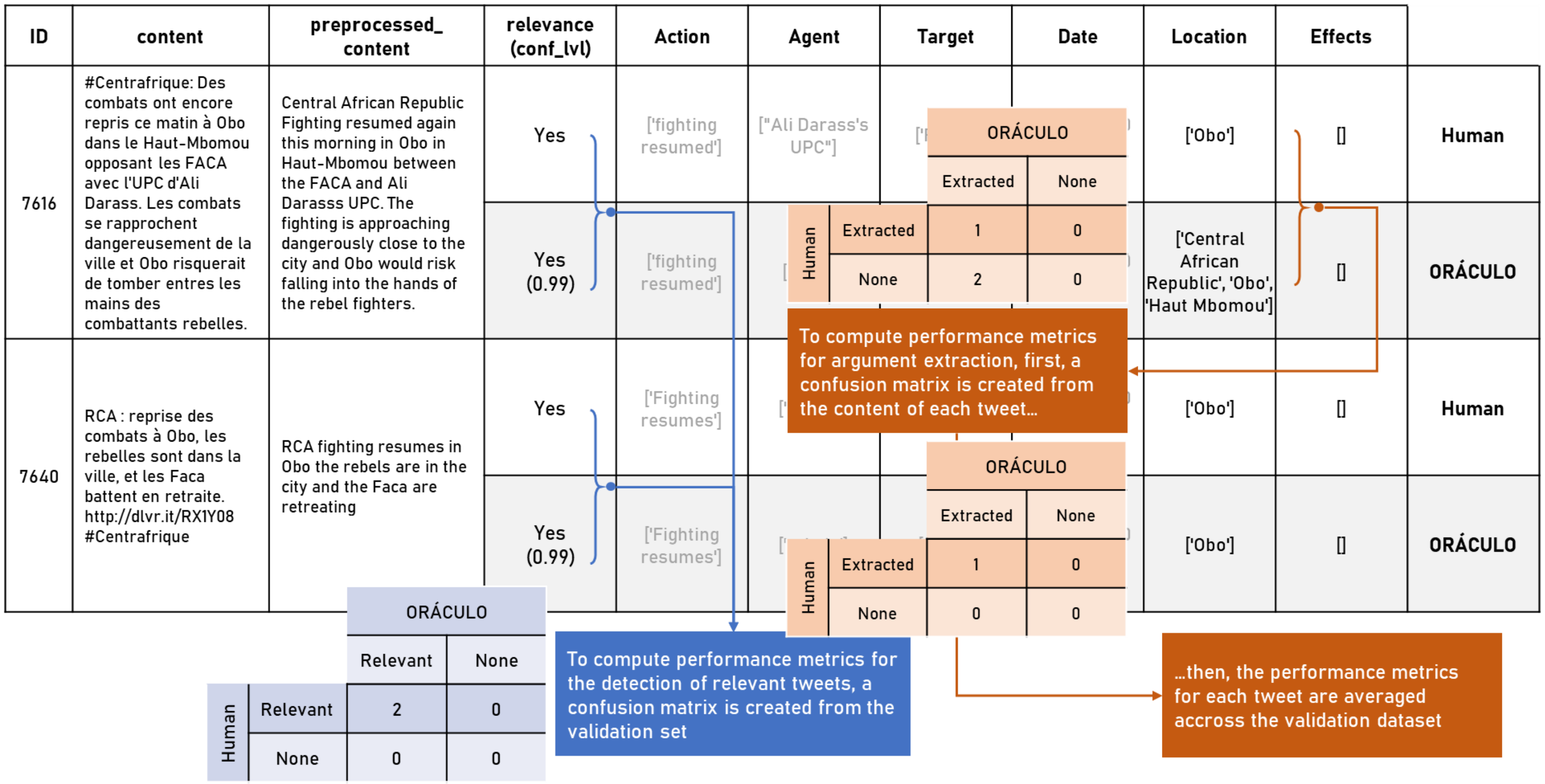
5.1.1 Snips NLU Event Extractor

The *training/validation dataset* was used to perform a stratified *n*-folds[35] cross-validation on the snips NLU event extractor. This manually classified dataset contains the same features as the *extracted event dataset* (Table 10, page 49) but includes all 8 518 scraped tweets (Figure 17, page 43). To allow cross-validation, it was split into *n* folds of 8 518/*n* tweets each (of which 11.3% were relevant, maintaining the same the same class ratio as the main dataset). Thus, *n* iterations of training and validation were conducted during each cross-validation test: when a given fold was used as the validation set, the other *n-1* were used as the training set. For each iteration, we compared the results (event detection and argument extraction) of the snips NLU event extractor with those of the manual extraction, computing Precision, Recall and F1 for each argument and for overall event detection, and Cohen's Kappa, Accuracy, Balanced Accuracy and AUROC for event detection alone (Appendix A: Performance Metrics). Figure 29 (page 63) illustrates the evaluation procedure.

We tested combinations of three strategies and tuned the values of two parameters to optimize the performance of the snips NLU event extractor:

- First, we tested 4-, 5- and 6-folds cross-validation in order to understand the impact of increasing or decreasing the amount of training data (first parameter).
- Then, to counteract the low amount of training data, we tested oversampling (*i.e.*, repeating) the relevant tweets in the training set, increasing effective training data.
- We tested lemmatizing the verbs in the training and validation datasets to improve the performance of the deterministic intent parser at extracting the *Action* argument.
- To further improve the deterministic intent parser, we also tested populating the entities in the training file with sets of known values for the *Action*, *Agent*, *Target*, *Location*, and *Effects* arguments (*i.e.*, creating a "gazetteer").
- Finally, once the best combination of strategies was found, we tweaked the relevance confidence level threshold (second parameter) in order to understand its effect on event detection.

[35] We selected folds and not splits to ensure that every tweet was parsed by the trained Snips NLU engine. Using 5 folds resulted in the usually recommended 80% training/20% validation split of the *training/validation dataset*. Cross-validation with 4 or 6 folds was also tested, but both options produced worse results than the 5-folds baseline.



ID	content	preprocessed_ content	relevance (conf_lvl)	Action	Agent	Target	Date	Location	Effects	
7616	#Centrafrique: Des combats ont encore repris ce matin à Obo dans le Haut-Mbomou opposant les FACA avec l'UPC d'Ali Darass. Les combats se rapprochent dangereusement de la ville et Obo risquerait de tomber entres les mains des combattants rebelles.	Central African Republic Fighting resumed again this morning in Obo in Haut-Mbomou between the FACA and Ali Darasss UPC. The fighting is approaching dangerously close to the city and Obo would risk falling into the hands of the rebel fighters.	Yes	['fighting resumed']	["Ali Darass's UPC"]			['Obo']	[]	Human
			Yes (0.99)	['fighting resumed']				['Central African Republic', 'Obo', 'Haut Mbomou']	[]	ORÁCULO
7640	RCA : reprise des combats à Obo, les rebelles sont dans la ville, et les Faca battent en retraite. http://dlvr.it/RX1Y08 #Centrafrique	RCA fighting resumes in Obo the rebels are in the city and the Faca are retreating	Yes	['Fighting resumes']				['Obo']	[]	Human
			Yes (0.99)	['Fighting resumes']				['Obo']	[]	ORÁCULO

		ORÁCULO	
		Extracted	None
Human	Extracted	1	0
	None	2	0

To compute performance metrics for argument extraction, first, a confusion matrix is created from the content of each tweet…

		ORÁCULO	
		Extracted	None
Human	Extracted	1	0
	None	0	0

		ORÁCULO	
		Relevant	None
Human	Relevant	2	0
	None	0	0

To compute performance metrics for the detection of relevant tweets, a confusion matrix is created from the validation set

…then, the performance metrics for each tweet are averaged accross the validation dataset

Figure 29. The evaluation procedure used to compute performance metrics for the snips NLU event extractor function (tweets and results are taken from those of test 11). “Human” rows refer to the results of the manual event extraction (the training/validation dataset), while “ORÁCULO” refers to the results of the snips NLU event extractor. Note how the confidence levels assigned by Snips NLU to the respective tweets are displayed in the “relevance (conf_lvl)” column to demonstrate their role. Also note how imperfect human coding procedures (coding “Obo” instead of “Central African Republic”, “Obo” and “Haut Mbomou”) impact the measured performance of ORÁCULO. Finally, keep in mind that though tweets 7616 and 7640 were previously used to illustrate training procedures (Figure 21), they were not part of the training set used in the iteration of the snips NLU event extractor that extracted the arguments shown above, *i.e.*, we did not have to cheat to produce these results!

Table 13. Detailed results of the snips NLU event extractor tests. The average standard deviation across all performance metrics is 0.025 (minimum: 0.01; maximum: 0.07; standard deviation: 0.01).

				STRATEGIES				EVENT DETECTION						
Test ID	Training set (effective)	Test set	Number of Folds	Gazetteer	Over-sampling	Lemma-tization	Confidence Level Threshold	Accuracy	F1	Precision (P)	Recall (R)	AUROC	Cohen's Kappa	Balanced Accuracy
1	771	1704	5	No	No	No	0.85	0.73	0.45	0.29	0.95	0.83	0.33	0.83
2	803	1420	**6**	No	No	No	0.85	0.73	0.44	0.28	0.94	0.82	0.32	0.82
3	723	**2129**	4	No	No	No	0.85	0.73	0.45	0.29	0.95	0.83	0.33	0.83
4	771	1704	5	**Yes**	No	No	0.85	0.72	0.44	0.28	0.96	0.83	0.32	0.83
5	771	1704	5	No	No	**Yes**	0.85	0.73	0.45	0.29	0.97	0.83	0.33	0.83
6	771	1704	5	**Yes**	No	**Yes**	0.85	0.73	0.44	0.29	0.97	0.83	0.33	0.83
7	**6814***	1704	5	No	**Yes**	No	0.85	0.75	0.47	0.31	0.97	0.84	0.35	0.84
8	**6814***	1704	5	**Yes**	**Yes**	No	0.85	0.74	0.46	0.30	**0.98**	0.84	0.35	0.84
9	**6814***	1704	5	No	**Yes**	**Yes**	0.85	0.74	0.46	0.30	**0.98**	0.84	0.35	0.84
10	**6814***	1704	5	**Yes**	**Yes**	**Yes**	0.85	0.73	0.45	0.29	**0.98**	0.84	0.33	0.84
11	**6814***	1704	5	**Yes**	**Yes**	No	**0.987**	**0.88**	**0.62**	**0.49**	0.86	**0.87**	**0.56**	**0.87**

ARGUMENT EXTRACTION																		
Action			Agent			Target			Date*			Location			Effects			Test ID
F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	
0.48	0.66	0.54	0.49	0.68	0.63	0.43	0.67	0.57	0.88	0.88	0.88	0.52	0.49	0.68	0.59	0.75	0.63	*1*
0.48	**0.67**	0.54	0.52	0.68	0.64	0.45	**0.68**	0.57	0.88	0.88	0.88	0.52	0.48	0.69	0.59	0.75	0.63	*2*
0.45	0.64	0.53	0.52	**0.70**	0.64	0.42	0.66	0.55	0.88	0.88	0.88	0.52	0.48	0.68	0.59	**0.77**	0.62	*3*
0.48	0.66	0.54	0.51	**0.70**	0.63	0.43	0.64	0.58	0.88	0.88	0.88	0.52	0.49	0.70	**0.60**	0.75	0.64	*4*
0.29	0.52	0.33	0.46	0.63	0.58	0.39	0.65	0.51	0.88	0.88	0.88	0.52	0.48	0.69	0.54	0.71	0.58	*5*
0.27	0.51	0.30	0.44	0.63	0.56	0.37	0.60	0.50	0.88	0.88	0.88	0.52	0.48	0.69	0.55	0.72	0.59	*6*
0.51	0.66	0.58	0.52	0.67	0.67	**0.46**	0.63	**0.62**	0.88	0.88	0.88	0.53	0.49	0.70	**0.60**	0.71	**0.65**	*7*
0.49	0.64	0.57	**0.53**	0.65	**0.68**	0.45	0.63	0.58	0.88	0.88	0.88	0.53	0.49	0.70	**0.60**	0.72	**0.65**	*8*
0.30	0.47	0.34	0.49	0.62	0.61	0.42	0.58	0.57	0.88	0.88	0.88	**0.54**	**0.50**	0.71	0.56	0.67	0.62	*9*
0.30	0.48	0.35	0.47	0.60	0.61	0.40	0.58	0.54	0.88	0.88	0.88	0.53	**0.50**	0.71	0.58	0.68	0.62	*10*
0.51	0.64	**0.59**	0.52	0.64	0.66	**0.46**	0.63	0.60	0.88	0.88	0.88	**0.54**	**0.50**	**0.72**	**0.60**	0.70	**0.65**	*11*

Above, Table 13 (page 64), showcases the complete snips NLU event extractor test results. Three main observations can be made regarding the effects of the strategies and parameters on extractor performance:

- **Small variations in the size of the effective training set produced no significant effects on performance**, *i.e.*, varying the number of folds had no observable effect. This is likely the effect of the deterministic intent parser, which uses the known entity values to directly extract event arguments and detect events, even with little training data.
- **Oversampling produced a weak positive effect on performance without negative tradeoffs.** Conversely, lemmatizing verbs in the tweet content significantly lowered performance, especially when extracting the *Action* argument. Surprisingly, the use of gazetteers was neutral or weakly positive at best, perhaps because the lists of known entity values were redundant. Even if the entities in the training file are defined using few values (recall Figure 21, page 48), during training those entities are populated automatically with the examples contained in the utterances, so the gazetteers of CAR Civil War factions, Mission Task Verbs, and CAR locations went mostly unneeded. (Conversely, the true value of the gazetteers might lie in allowing an instance of snips NLU event extractor trained on a specific context to quickly adapt to a difference set of circumstances.)
- **Increasing the relevance confidence level threshold greatly increased precision and F1 at detecting events, but also lowered recall.** By deploying the best combination of strategies (*oversampling* and *gazetteer*) and increasing the relevance confidence level threshold, Test 11 achieved by far the best results at detecting events from a conventional event extraction perspective. However, lowering recall means that important intelligence can be more easily lost, meaning that increasing the relevance confidence level threshold cannot be relied upon to increase the performance of ORÁCULO. Therefore, the configuration used on Test 8 (*oversampling* and *gazetteer*) achieved the highest performance from an intelligence standpoint.

Finally, we compared the event detection performance of the two best snips NLU event extractor configurations (Test 8 and Test 11) with that of Logistic Regression and SVM classifiers. Though these classifiers can only perform event detection – a binary classification problem –, comparing them with snips NLU event extractor allows us to gauge the strengths and weaknesses of the latter. Thus, the Logistic Regression and SVM classifiers were trained and tested using their default parameters on the same *training/validation dataset*. TF-IDF vectorization was used to convert the preprocessed tweets into 12 656-feature vectors, as both classifiers require numeric vectors as input. Below, Table 14 presents the results.

Table 14. Comparison between the event detection performance of the two best configurations of the snips NLU event extractor and the Logistic Regression and SVM classifiers. All models were trained and tested on the *training/validation dataset* using 5-fold cross-validation. The average standard deviation across the performance metrics of the two classifiers is 0.025 (minimum: 0.004; maximum: 0.05; standard deviation: 0.01).

	EVENT DETECTION							
Model	**Vector**	**Acc.**	**F1**	**P**	**R**	**AUROC**	**Kappa**	**Bal. Acc**
Snips NLU (Test 8)	Word2Vec (internal)	0.74	0.46	0.30	**0.98**	**0.84**	0.35	**0.84**
Snips NLU (Test 11)	Word2Vec (internal)	0.88	0.62	0.49	**0.86**	**0.87**	0.56	**0.87**
Logistic Regression	TF-IDF	**0.94**	**0.66**	**0.93**	0.51	0.75	**0.63**	0.75
Support Vector Machine	TF-IDF	**0.95**	**0.75**	**0.92**	0.63	0.81	**0.72**	0.81

Though the Logistic Regression and SVM scored higher on Accuracy, F1, Precision and on Cohen's Kappa, they achieved worse Recall, AUROC, and Balanced Accuracy than the snips NLU event extractor. Again, though they perform better from a conventional event detection perspective, if Logistic Regression and SVM classifiers were deployed as the event detectors of ORÁCULO, their low Recall would jeopardize the task of detecting all possible conflict-related events reported by the Twitter accounts of selected online news sources, reducing even further the number of extracted events. Therefore, test results validate the choice of Snips NLU as the engine of the Event Extractor, as its high Recall allows the detection and extraction of the largest possible number of events, and its low Precision can be compensated by analyst supervision.

5.1.2 Geocoder

To assess the performance of the geocoder function at selecting and geocoding the *event location*, we ran the function on the 965 events in the *training/validation dataset* and compared the result with that of the manual classification, assessing events as either correctly or incorrectly geocoded. We found that despite the use of a heuristic to decide the final *event location*, the geocoder correctly discovered the *event location* of 924 tweets, resulting in **a geocoding accuracy of 0.959**.

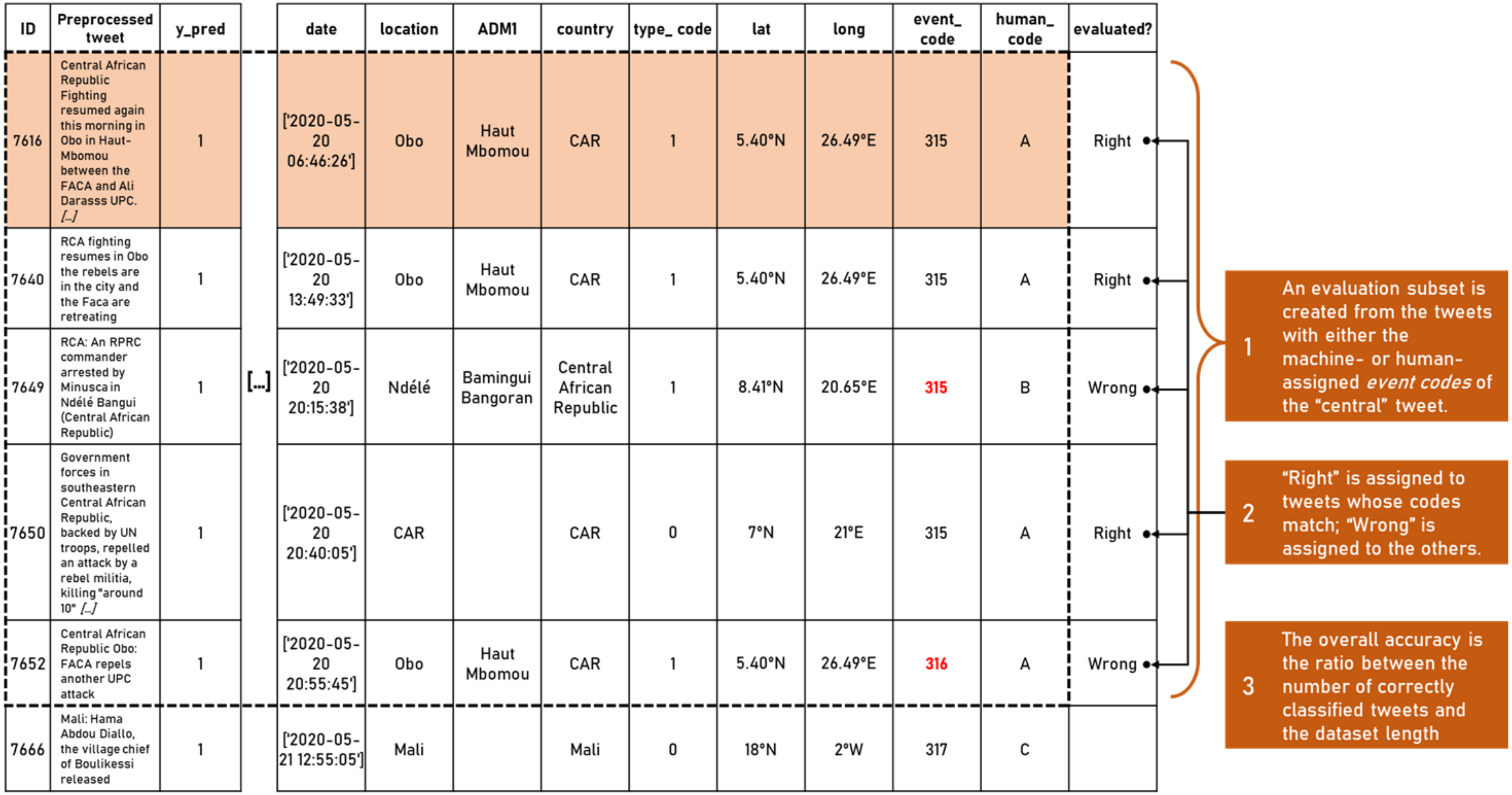
5.1.3 Event Merger

As with the geocoder function, the performance of the event merger was obtained by comparing the event codes assigned manually to the 965 events in the *training/validation dataset* with the event codes assigned by the event merger. To compare human and automatic event merging, for each event in the *training/validation dataset*, we selected a subset of the dataset (the "evaluation subset") comprising events with the same "human" event code and with the same "machine" event code as the "central" (selected) event. For each event of that subset, if the "human" and "machine" event codes were the same as those of the "central event", they were counted as correctly classified. The overall accuracy of the event merging was the ratio between the number of correctly merged events and the total number of events in the *training/validation dataset* (Figure 30, page 68). An **overall event merging accuracy of 0.812** was achieved.

5.1.4 Event Extractor

To assess the potential of ORÁCULO for event extraction, we compared the *ACLED event dataset* with the geocoded, merged, and refined results of the best snips NLU event extractor configuration, Test 8 (Table 13), to discover their amount of mutual information. The following steps were necessary to generate the *Test 8 refined event dataset*:

- First, we selected the best snips NLU event extractor test. Test 8 was selected because it achieved the not only the highest event detection recall, but because its performance metrics for argument extraction were generally on par with the best results for each argument. The output of Test 8 contained 940 tweets with relevant events (out of a total of 965), but also contained 2163 irrelevant tweets due to the low precision of the Test 8 configuration.



ID	Preprocessed tweet	y_pred		date	location	ADM1	country	type_ code	lat	long	event_ code	human_ code	evaluated?
7616	Central African Republic Fighting resumed again this morning in Obo in Haut-Mbomou between the FACA and Ali Darasss UPC. *[…]*	1		['2020-05-20 06:46:26']	Obo	Haut Mbomou	CAR	1	5.40°N	26.49°E	315	A	Right
7640	RCA fighting resumes in Obo the rebels are in the city and the Faca are retreating	1		['2020-05-20 13:49:33']	Obo	Haut Mbomou	CAR	1	5.40°N	26.49°E	315	A	Right
7649	RCA: An RPRC commander arrested by Minusca in Ndélé Bangui (Central African Republic)	1	[…]	['2020-05-20 20:15:38']	Ndélé	Bamingui Bangoran	Central African Republic	1	8.41°N	20.65°E	315	B	Wrong
7650	Government forces in southeastern Central African Republic, backed by UN troops, repelled an attack by a rebel militia, killing "around 10" *[…]*	1		['2020-05-20 20:40:05']	CAR		CAR	0	7°N	21°E	315	A	Right
7652	Central African Republic Obo: FACA repels another UPC attack	1		['2020-05-20 20:55:45']	Obo	Haut Mbomou	CAR	1	5.40°N	26.49°E	316	A	Wrong
7666	Mali: Hama Abdou Diallo, the village chief of Boulikessi released	1		['2020-05-21 12:55:05']	Mali		Mali	0	18°N	2°W	317	C	

1 An evaluation subset is created from the tweets with either the machine- or human-assigned *event codes* of the "central" tweet.

2 "Right" is assigned to tweets whose codes match; "Wrong" is assigned to the others.

3 The overall accuracy is the ratio between the number of correctly classified tweets and the dataset length

Figure 30. Diagram of the evaluation procedure used to compute the accuracy of the event merger function. Note that events 7649 and 7652 are incorrectly classified on purpose to illustrate the evaluation procedure. Also note the use of an *evaluated?* feature to allow the computation of the merging accuracy.

- Then, we aggregated the results of each iteration of Test 8 into a single *extracted event dataset* and fed it to the geocoder and event merger functions. The result was a *Test 8 merged event dataset* that contained 620[36] unique events, but also 1 207 irrelevant events extracted as false positives. When considering CAR events only, the dataset contained 238 relevant events and 322 false positives.
- Finally, we manually supervised the merged event dataset to simulate analyst supervision, removing false positives and filtering out non-CAR events and CAR events which did not extract an event location beyond "CAR" itself. The result was a *Test 8 refined event dataset* comprising 202 CAR Civil War events.

After generating the *Test 8 refined event dataset*, we compared it to the *ACLED event dataset*. To do so, both datasets were concatenated and re-sorted in a joint table, and joint event codes were manually assigned to ACLED and ORÁCULO events that describe the same real-world event. Then, events with the same joint code were merged to form an evaluation table, which describes how many real-world events were detected by ACLED, by ORÁCULO, or by both. Finally, the resulting confusion matrix was used to compute the Normalized Mutual Information score (NMI), which evaluates the degree of mutual information between the datasets. The result was that the ***Test 8 refined event dataset*** **and the** ***ACLED event dataset*** **achieved an NMI score of 0.395.** Below, Figure 31 uses a Venn diagram to illustrate the amount of mutual information in the two test datasets, while Figure 32 (page 70) describes the comparison procedure.

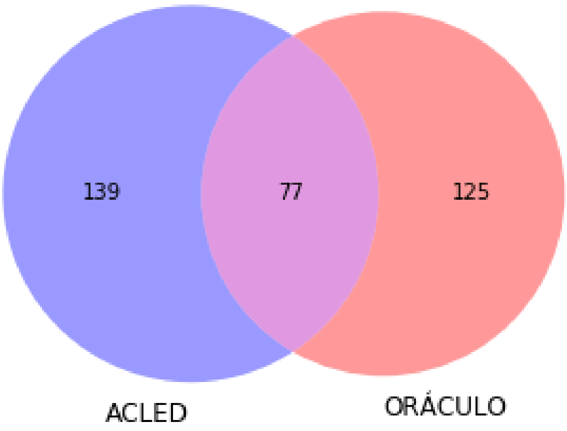
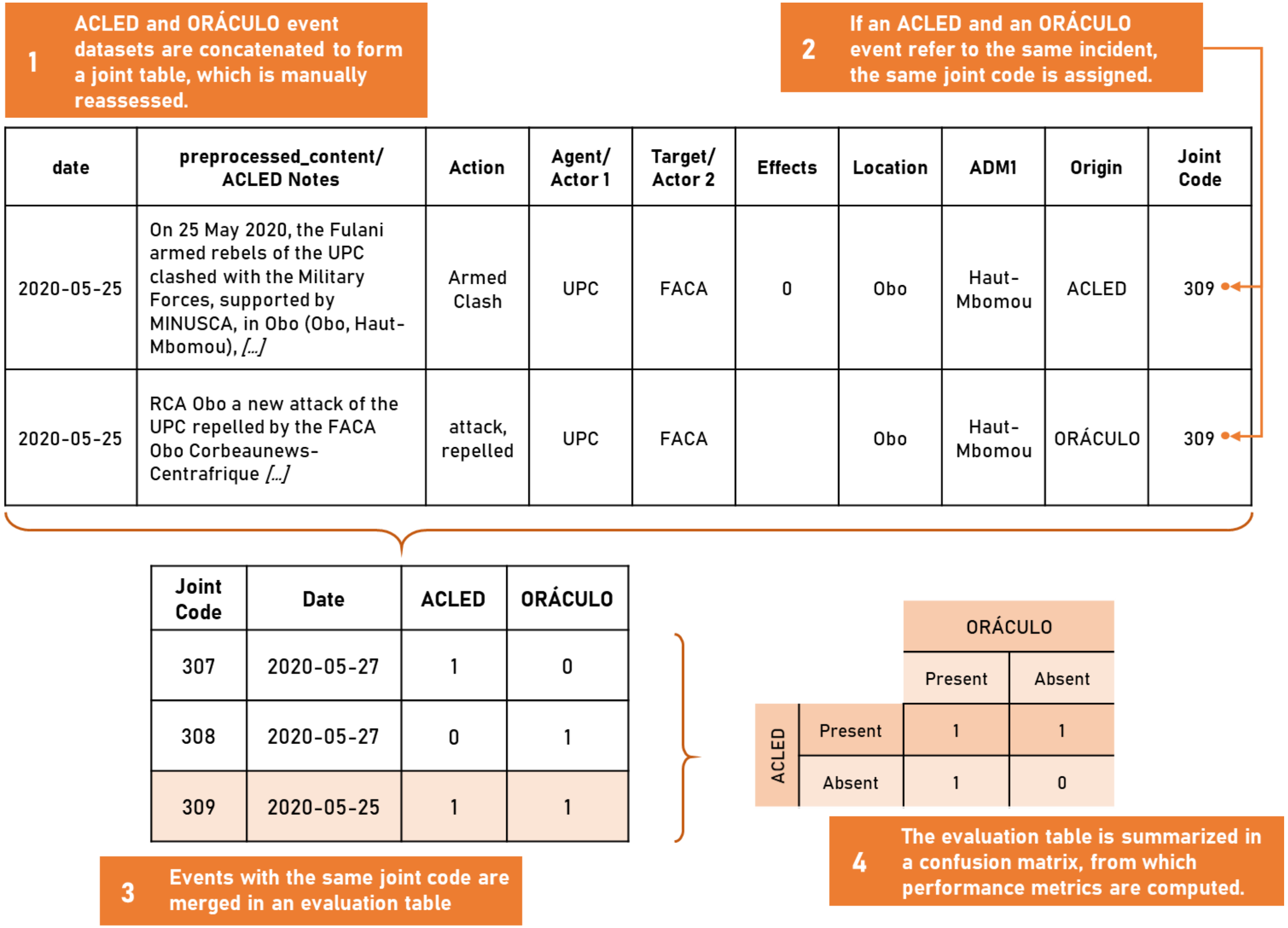


Figure 31. Venn diagram of the *ACLED event dataset* ("ACLED") and *Test 8 refined event dataset* ("ORÁCULO"). Note how only 216 unique ACLED events (out of 222) were used in the comparison: several ACLED events reported the same overall real-world event, so they were manually merged to allow direct comparison between the ACLED and ORÁCULO datasets.

[36] Some relevant tweets were not correctly merged into unique events, so there are more events in Test 8 merged event dataset (620) than their true number (615).



1 ACLED and ORÁCULO event datasets are concatenated to form a joint table, which is manually reassessed.

2 If an ACLED and an ORÁCULO event refer to the same incident, the same joint code is assigned.

date	preprocessed_content/ ACLED Notes	Action	Agent/ Actor 1	Target/ Actor 2	Effects	Location	ADM1	Origin	Joint Code
2020-05-25	On 25 May 2020, the Fulani armed rebels of the UPC clashed with the Military Forces, supported by MINUSCA, in Obo (Obo, Haut-Mbomou), *[...]*	Armed Clash	UPC	FACA	0	Obo	Haut-Mbomou	ACLED	309
2020-05-25	RCA Obo a new attack of the UPC repelled by the FACA Obo Corbeaunews-Centrafrique *[...]*	attack, repelled	UPC	FACA		Obo	Haut-Mbomou	ORÁCULO	309

Joint Code	Date	ACLED	ORÁCULO
307	2020-05-27	1	0
308	2020-05-27	0	1
309	2020-05-25	1	1

3 Events with the same joint code are merged in an evaluation table

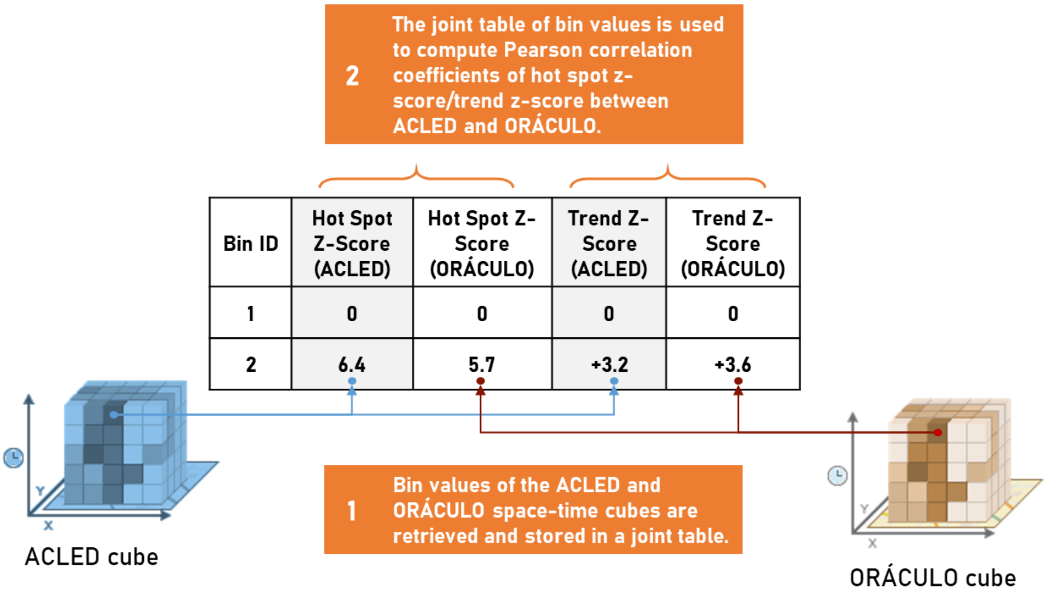
		ORÁCULO	
		Present	Absent
ACLED	Present	1	1
	Absent	1	0

4 The evaluation table is summarized in a confusion matrix, from which performance metrics are computed.

Figure 32. Test procedure used to compare the ACLED and ORÁCULO event datasets.

5.1.5 Hot Spots Detector

The potential of ORÁCULO for hot spots detection was assessed by computing the Pearson Correlation Coefficient between ACLED and ORÁCULO space-time cubes. To do so, we used the geodatabase generator and emerging hots spots detector functions to generate event space-time cubes from the *ACLED event dataset* and the *Test 8 refined event dataset* across a set of possible cube parameters. Then, we computed the Pearson Correlation Coefficient of the *emerging hot spots count z-score* and *trend count z-score* between the resulting cubes. Below, Figure 33 illustrates the comparison procedure.



2 The joint table of bin values is used to compute Pearson correlation coefficients of hot spot z-score/trend z-score between ACLED and ORÁCULO.

Bin ID	Hot Spot Z-Score (ACLED)	Hot Spot Z-Score (ORÁCULO)	Trend Z-Score (ACLED)	Trend Z-Score (ORÁCULO)
1	0	0	0	0
2	6.4	5.7	+3.2	+3.6

1 Bin values of the ACLED and ORÁCULO space-time cubes are retrieved and stored in a joint table.

Figure 33. Test procedure used to compare the ACLED and ORÁCULO event space-time cubes. Note that direct comparison is only possible when both cubes use the same set of parameters and the cover the same area and time period.

Possible cube parameters were obtained from two sources: the default ArcGIS methods and the analysis of the spatial and temporal autocorrelations of the event feature classes of the event geodatabases. Regarding the default methods, while ESRI (2020b) (2020a) recommends selecting cube parameters based on the spatial and temporal context of the phenomena, their tools provide methods to discover parameters from the distribution of the input features if no parameters are specified:

- The “Create Space-Time Cube By Aggregating Points” tool determines spatial bin size by computing the average distance to the nearest neighbor of the input point feature class, while the temporal bin size is determined using a set of algorithms developed by Shimazaki and Shinomoto (2007), which take the spike count in the input time series as their only input.

- The “Emerging Hot Spot Analysis” tool discovers spatial neighborhood size by computing the bandwidth of the 2-dimensional visualization of the space-time cube (equivalent to aggregating events using a square grid) using an adaptation of Silverman's Rule-of-thumb bandwidth estimation formula **(3)**. For the temporal neighborhood, when none is provided, the temporal neighborhood size is set to one time step.

$$SpatialNei = \begin{cases} 0.9 \times SD \times n^{-0.2} \ \ if \ SD < \sqrt{\frac{1}{\ln(2)}} \times D_m \\ 0.9 \times \sqrt{\frac{1}{\ln(2)}} \times D_m \times n^{-0.2} \ if \ SD > \sqrt{\frac{1}{\ln(2)}} \times D_m \end{cases} \tag{3.1}$$

$$SD = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{X})^2}{n} + \frac{\sum_{i=1}^{n}(y_i - \bar{Y})^2}{n}} \tag{3.2}$$

The spatial neighborhood/bandwidth of a point feature class with *n* points is given by equations **(3.1)** and **(3.2)**, where:

D_m represents the median distance from the mean spatial center of the point feature class;
SD represents the Standard Distance of the point feature class;
x_i and y_i represent the spatial coordinates of feature *i;*

Source: (ESRI, 2020e).

The other method used to discover possible cube parameters consists in aggregating the event datasets using various grid sizes and time intervals and computing their spatial and temporal autocorrelations. For a given grid size/time interval, if significant autocorrelation of event counts exists up to any space/time lag, then that grid size/time interval can be aggregated with the autocorrelated lags, creating a larger homogenous space/time region. Conversely, if no autocorrelation exists at any lag, then the grid size/time interval represents a viable spatial/temporal bin size. This need to create large homogenous regions is balanced by the need to discover the smallest possible homogenous regions, as large space-time cube bins can “dissolve” event counts per bin, *i.e.*, the event count in a given bin will be close to the mean event count per bin across the dataset, “hiding” event hot spots (recall equation **(1)**, page 55).

Thus, to discover the smallest possible homogenous spatial bins, we aggregated the event datasets using 5-, 10-, 15-, 20- and 25-kilometer square grids and computed their spatial autocorrelation. The results are shown in Table 15 (page 73). Since both event datasets exhibited significant spatial autocorrelation *at least* up to the first spatial lag when aggregated with the 5- and 10-kilometer grids, they were discarded from the set of possible cube parameters. For the remaining grid sizes, even though the *ACLED event dataset* only stopped exhibiting any significant autocorrelation at the 25-kilometer grid while the Test 8 dataset stopped at the 15-kilometer grid[37], the need to find common cube parameters led us to include 15-, 20-, and 25-kilometer grids in the set of possible values.

Table 15. Spatial autocorrelation (Moran's I) (Moran, 1950), z-score and confidence level for the first three distance bands of the event datasets aggregated using 5-, 10-, 15-, 20- and 25-kilometer grids. The last distance band of the largest grid at which significant[38] ($p < 0.05$) spatial autocorrelation exists is highlighted in bold.

		ACLED event dataset (216 events)		Test 8 *refined event dataset* (202 events)	
Grid Size	Distance Band	Moran's I	z-score	Moran's I	z-score
5-km grid	5 km	0.042	10.635***	0.009	2.296*
	10 km	0.020	8.713***	0.003	1.132
	15 km	0.010	6.830***	0.003	2.140*
10-km grid	10 km	0.050	6.371***	**0.019**	**2.457***
	20 km	0.020	4.691***	0.006	1.333
	30 km	0.010	3.230**	0.001	0.442
15-km grid	15 km	0.038	3.356***	0.008	0.687
	30 km	0.012	1.845	0.005	0.860
	45 km	0.003	0.835	-0.000	0.017
20-km grid	20 km	**0.031**	**2.092***	0.002	0.154
	40 km	0.009	1.071	-0.003	-0.267
	60 km	0.002	0.472	-0.002	-0.190
25-km grid	25 km	-0.004	-0.151	-0.009	-0.410
	50 km	-0.007	-0.523	-0.010	-0.790
	75 km	-0.005	-0.580	-0.003	-0.310

Possible temporal bin sizes were discovered using the same process: the event datasets were aggregated using 1-day, 1-week, and 1-month time intervals, and their temporal autocorrelation was computed. Only the events after 04 November 2019 are considered, since up to this date there is a large difference between the event counts of the ACLED and ORÁCULO (Test 8) event datasets (recall Figure 19, page 45). Below, Table 16 (page 74) presents temporal autocorrelation results. While neither event dataset exhibits autocorrelation at any bin size, the 1-day time interval was excluded from the set of possible cube parameters, since it does not account for the uneven coverage of the online new sources. Finally, further below, Table 17 (page 74) summarizes the set of possible values obtained by the two methods plus the heuristically determined values for the spatial and temporal neighborhood.

[37] The differences in autocorrelation observed in Table 15 between the Test 8 *refined event dataset* and the *ACLED event dataset* are consistent with the differences in spatial distribution that can be observed in Figure 20 (page 13).

[38] * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table 16. Temporal autocorrelation for the first three lags of the 1-day, 1-week and 1-month time series. No significant autocorrelation was found.

		ACLED event dataset (>2019-11-04)	**Test 8 *refined event dataset*** (>2019-11-04)
Temporal Bin	Time Lag	Autocorrelation (Pearson)	Autocorrelation (Pearson)
1-day	1 day	0.138	-0.021
	2 days	-0.037	0.078
	3 days	0.084	0.072
1-week	1 week	-0.148	-0.015
	2 weeks	-0.071	0.062
	3 weeks	-0.115	0.206
1-month	1 month	-0.557	0.342
	2 months	0.263	0.058
	3 months	-0.138	-0.149

Table 17. Set of possible cube parameters used in the parameter sweep. In total, 120 tests were performed.

	Origin		
	ArcGIS methods	**Autocorrelation**	**Heuristic**
bin shape	Square (Fishnet) grid Hexagonal grid		
spatial bin size	300 km	15 km 20 km 25 km	
temporal bin size	1 Month	1 Week	
spatial neighborhood size	*300 km bin:* 300 km *15 km bin:* 72 km *20 km bin:* 81 km *25 km bin:* 89 km		1 × spatial bin size 3 × spatial bin size
temporal neighborhood size	1 × temporal bin size		3 × temporal bin size 5 × temporal bin size

The next pages display the results of the emerging hot spots detector parameter sweep across the set of possible parameters summarized by Table 17. In total, 120 combinations of parameters were tested, and Table 18 (page 75) displays the top and bottom 10 results when sorted by "average correlation" – the average of *emerging hot spots count z-score* correlation and *trend count z-score* correlation. Then, Figure 34 (page 76) to Figure 41 (page 79) display the 2- and 3-dimensional visualizations (maps and scenes) of two selected tests, test 20 and test 35, while Table 19 (page 80) explains the hot spots categories used by ArcGIS to describe their temporal trends. While neither test 20 nor test 35 scored the highest average correlation, test 20 scored the highest *emerging hot spots count z-score* correlation overall, and test 35 scored the highest average correlation amongst the tests with usable spatial bin sizes (*i.e.*, not overly large 300-kilometer bins). Therefore, the hot spots detected in tests 20 and 35 should be the closest to the true CAR Civil War event hot spots *and* the closest to being actionable intelligence.

Table 18. Top and bottom 10 results (sorted by average correlation) of the emerging hot spots detector parameter sweep. The best values for each metric are highlighted in bold. The 2- and 3-dimensional visualizations of the tests shaded gray are shown in the next pages.

	Test ID	Bin shape	Spatial bin size	Temporal bin size	Spatial neighborhood size	Temporal neighborhood size (# bins)	*emerging hot spots count z-score* correlation (Pearson)	*trend count z-score* correlation (Pearson)	Average correlation (hot spots and trend)
Top 10	5	Fishnet	300 km	1 months	300 km	5	0.629***	0.787***	**0.708**
	65	Hexagonal	300 km	1 months	300 km	5	0.602***	**0.790*****	0.696
	35	Fishnet	20 km	1 months	82 km	5	0.670***	0.673***	0.672
	53	Fishnet	25 km	1 months	89 km	5	0.660***	0.683***	0.672
	113	Hexagonal	25 km	1 months	89 km	5	0.663***	0.680***	0.671
	17	Fishnet	15 km	1 months	72 km	5	0.675***	0.663***	0.669
	77	Hexagonal	15 km	1 months	72 km	5	0.670***	0.665***	0.667
	20	Fishnet	15 km	1 months	15 km	5	**0.685*****	0.650***	0.667
	98	Hexagonal	20 km	1 months	20 km	5	0.681***	0.644***	0.663
	119	Hexagonal	25 km	1 months	75 km	5	0.667***	0.656***	0.661
Bottom 10	64	Hexagonal	300 km	1 months	300 Km	3	0.592***	0.490***	0.541
	69	Hexagonal	15 km	1 weeks	15 Km	1	0.500***	0.576***	0.538
	9	Fishnet	15 km	1 weeks	15 Km	1	0.500***	0.563***	0.532
	3	Fishnet	300 km	1 months	300 Km	1	0.583***	0.466***	0.525
	1	Fishnet	300 km	1 weeks	300 Km	3	0.515***	0.506***	0.510
	63	Hexagonal	300 km	1 months	300 Km	1	0.577***	0.336***	0.456
	62	Hexagonal	300 km	1 weeks	300 Km	5	0.532***	0.348***	0.440
	0	Fishnet	300 km	1 weeks	300 Km	1	**0.473*****	0.388***	0.430
	61	Hexagonal	300 km	1 weeks	300 Km	3	0.513***	0.317***	0.415
	60	Hexagonal	300 km	1 weeks	300 Km	1	0.482***	**0.285*****	**0.384**

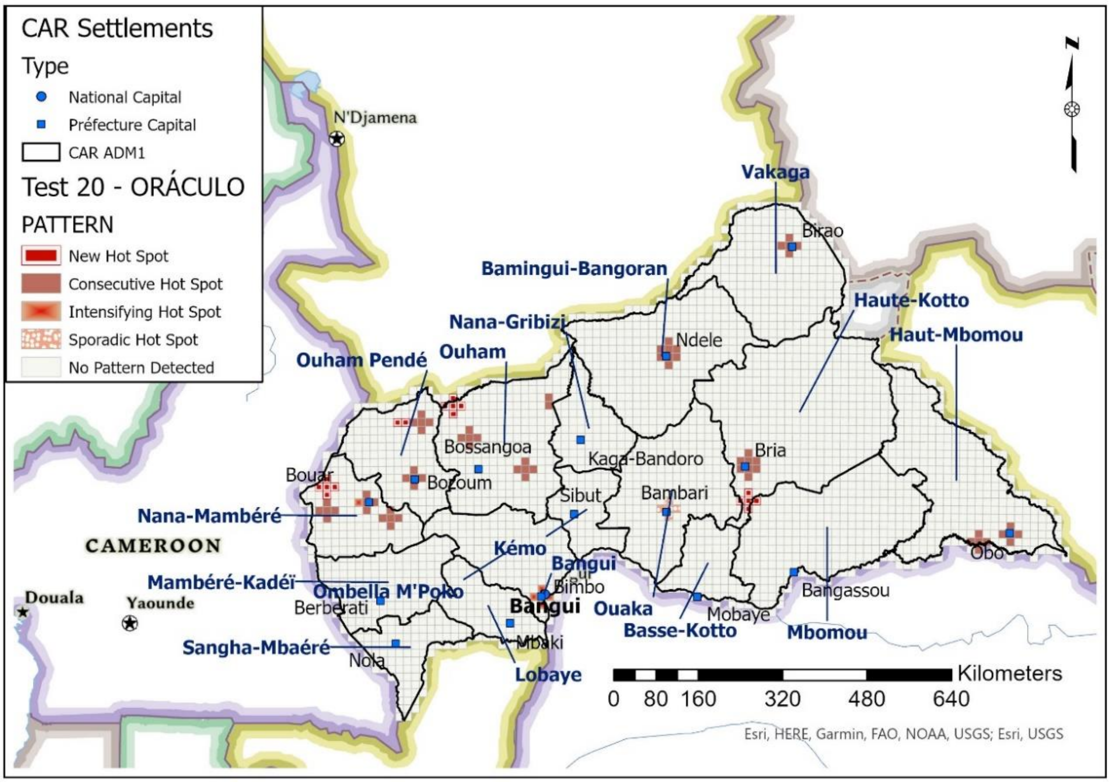


Figure 34. 2-dimensional visualization of the ORÁCULO *event space-time cube* generated in Test 20 (15-kilometers, 1-Month bins; 15-kilometers, 5 Months neighborhood).

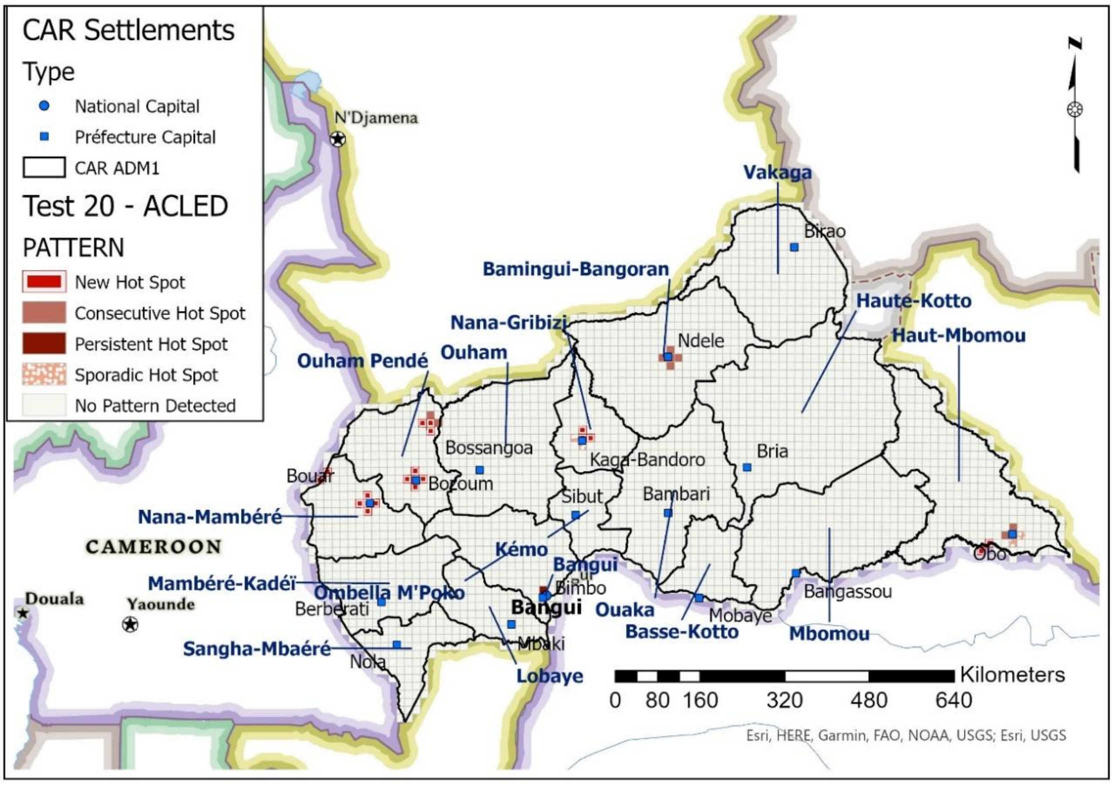
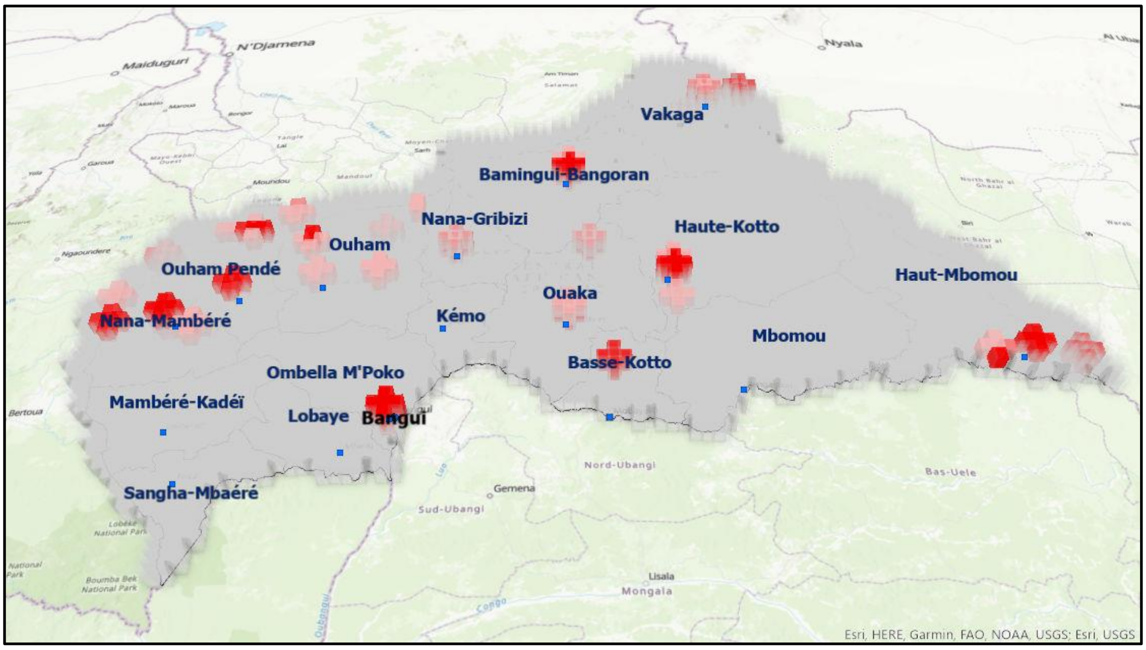
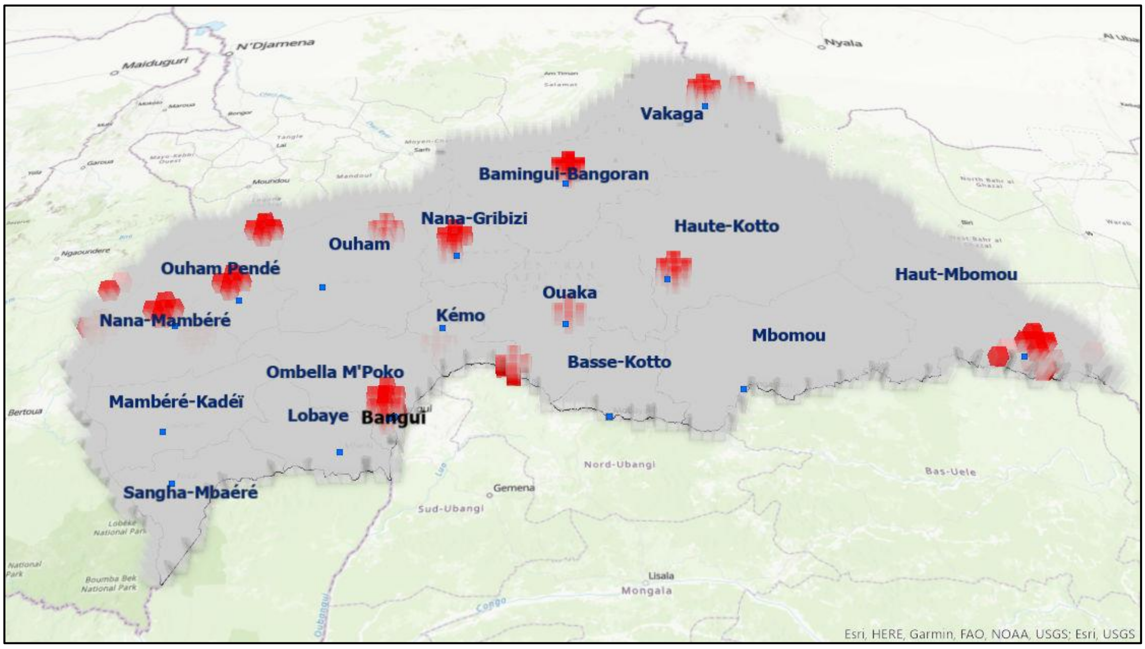


Figure 35. 2-dimensional visualization of the *ACLED event space-time cube* generated in Test 20 (15-kilometers, 1-Month bins; 15-kilometers, 5 Months neighborhood).



Esri, HERE, Garmin, FAO, NOAA, USGS; Esri, USGS

Figure 36. 3-dimensional visualization of the ORÁCULO *event space-time cube* generated in Test 20 (15-kilometers, 1-Month bins; 15-kilometers, 5 Months neighborhood).



Esri, HERE, Garmin, FAO, NOAA, USGS; Esri, USGS

Figure 37. 3-dimensional visualization of the *ACLED event space-time cube* generated in Test 20 (15-kilometers, 1-Month bins; 15-kilometers, 5 Months neighborhood).

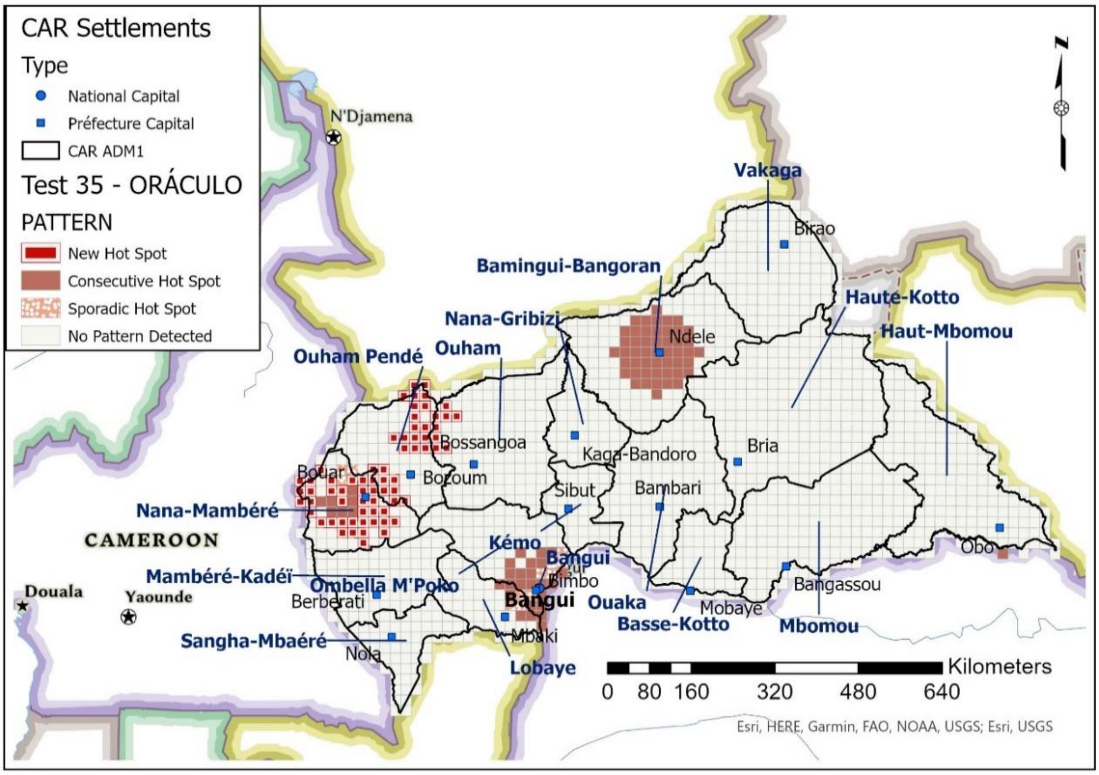


Figure 38. 2-dimensional visualization of the ORÁCULO *event space-time cube* generated in Test 35 (20-kilometers, 1-Month bins; 82-kilometers, 5 Months neighborhood).

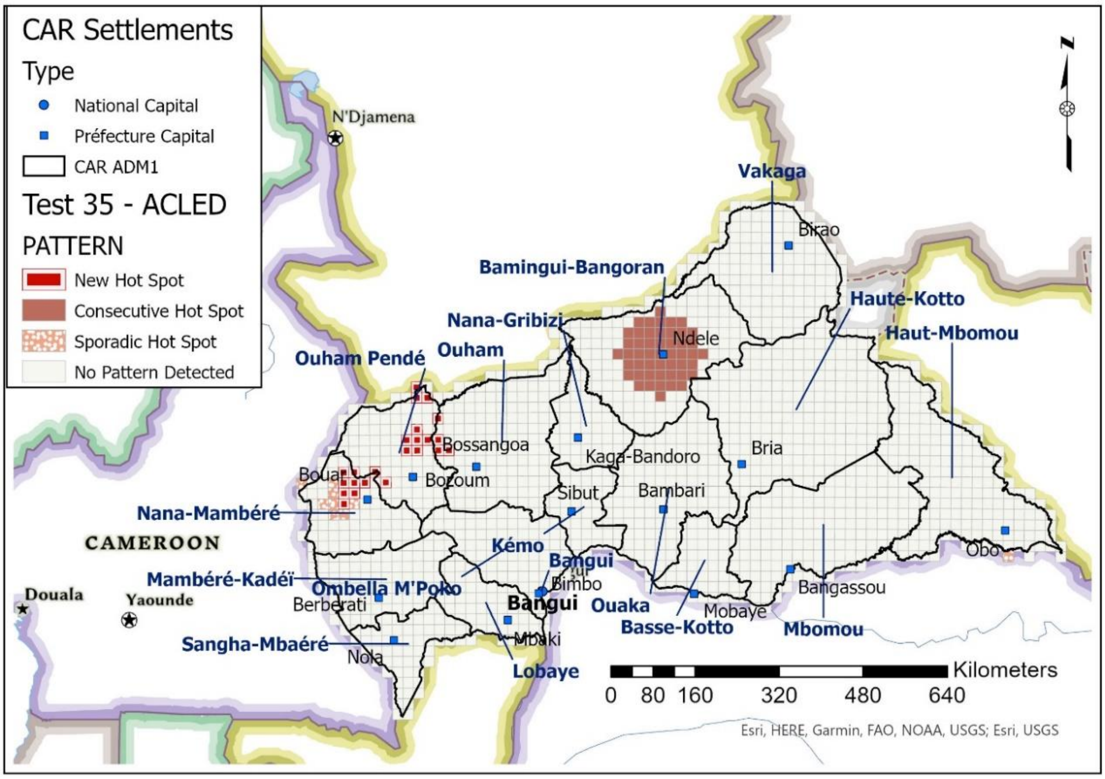


Figure 39. 2-dimensional visualization of the *ACLED event space-time cube* generated in Test 35 (20-kilometers, 1-Month bins; 82-kilometers, 5 Months neighborhood).

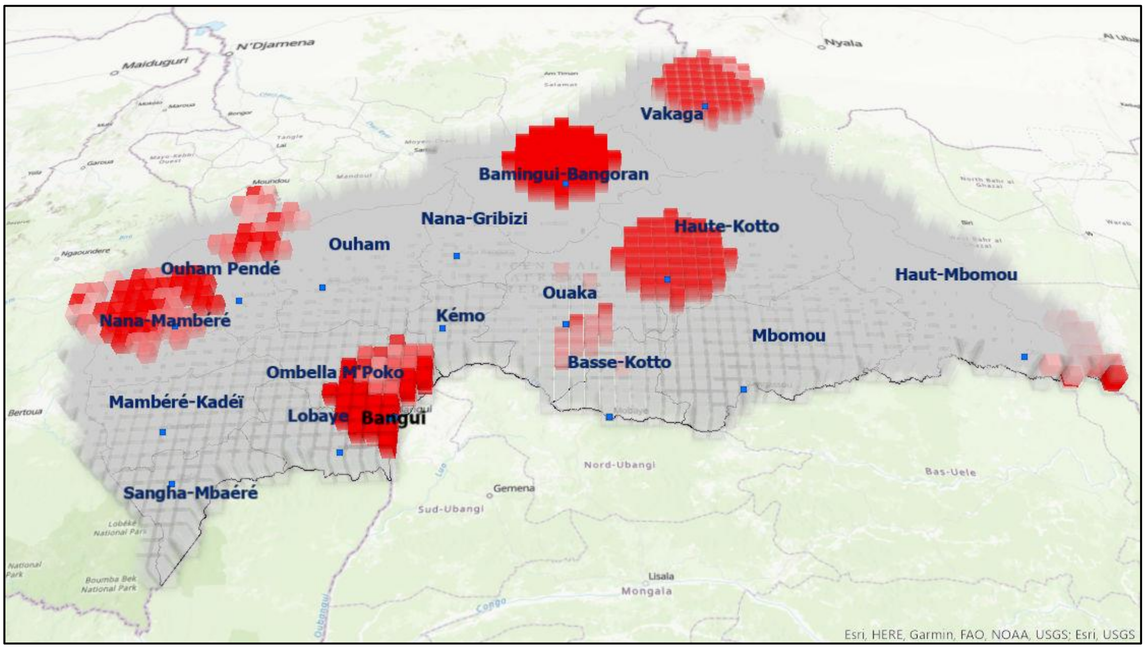


Figure 40. 3-dimensional visualization of the ORÁCULO *event space-time cube* generated in Test 35 (20-kilometers, 1-Month bins; 82-kilometers, 5 Months neighborhood).

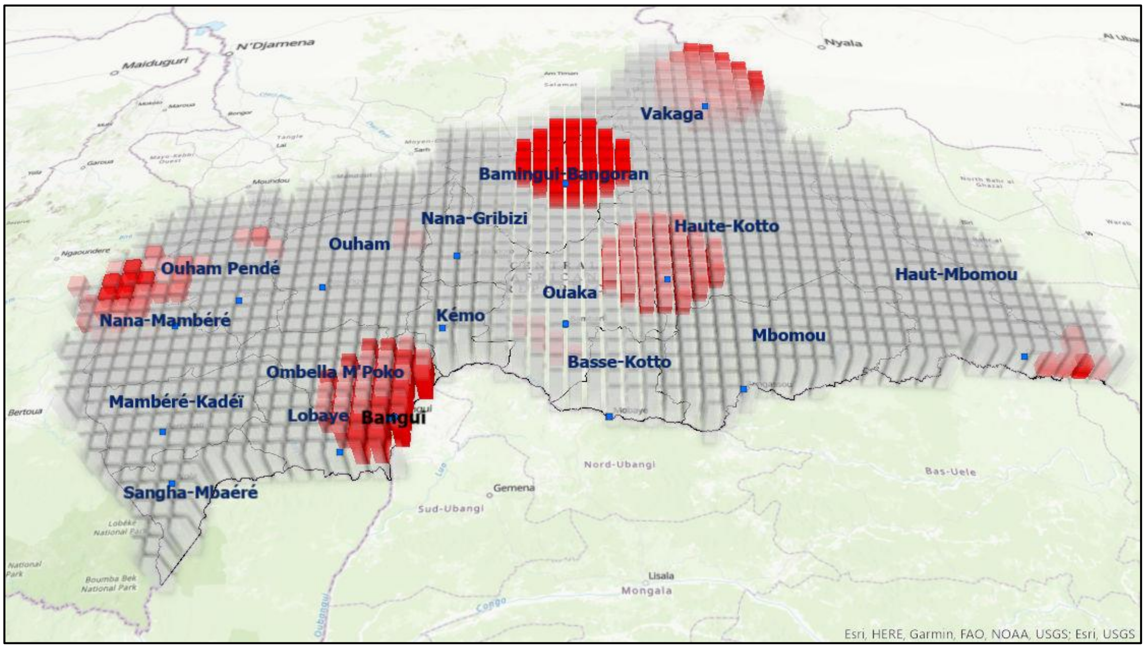
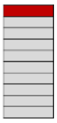
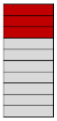
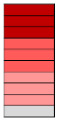
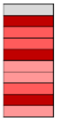
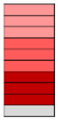
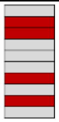
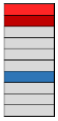
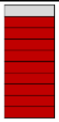
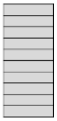


Figure 41. 3-dimensional visualization of the *ACLED event space-time cube* generated in Test 35 (20-kilometers, 1-Month bins; 82-kilometers, 5 Months neighborhood).

Table 19. Hot Spot categories according to the ESRI (2020a) system of classification. For each category, the "diagram" illustrates the type of cube bins that a cube column needs to have for it to be classified as the category. Red bins indicate a hot spot; the lighter the red, the weaker the *emerging hot spots count z-score*.

Pattern name	Final Time Step	Previous Time Steps	Hot Spots Percentage	Diagram
New Hot Spot	Hot Spot (z-score > 0)	No hot spots.		
Consecutive Hot Spot	Hot Spot (z-score > 0)	Consecutive hot spots up to final time step, but never had hot spots before that series.	<90%	
Intensifying Hot Spot	Hot Spot (z-score > 0)	Hot Spot z-score is increasing.	>90%	
Persistent Hot Spot	Hot Spot or no pattern	No trend in z-score intensity.	>90%	
Diminishing Hot Spot	Hot Spot (z-score > 0)	Hot Spot z-score is decreasing	>90%	
Sporadic Hot Spot	Hot Spot or no pattern	Some bins are hot spots.	<90%	
Oscillating Hot Spot	Hot Spot (z-score > 0)	Was a cold spot (z-score < 0) in at least one previous time step.	<90%	
Historical Hot Spot	No pattern	Most bins are hot spots.	>90%	
No Pattern Detected	No pattern	No hot spots.		

Given the results presented above, three main assessments can be made regarding the effect of the different parameters on the performance metrics and overall output:

- **Bin shape and spatial bin size have little influence on the *emerging hot spots count z-score* and *trend count z-score* correlations and on the location of the detected hot spots.** However, the *emerging hot spots count z-score* correlation tends to decrease when bin size increases by an order of magnitude: eight out of the bottom ten test results were obtained with a bin size of 300 Kilometers, while tests 5 and 65 scored significantly worse on *emerging hot spots count z-score* correlation than other tests using the same temporal parameters.

- **Temporal bin size and temporal neighborhood size have a large influence on the *emerging hot spots count z-score* and *trend count z-score* correlations and on the detection of long-term hot spots.** Both input datasets (ACLED: 216 events; ORÁCULO, Test 8: 202 events) are too sparse for 1-week aggregation and for small temporal neighborhoods, since they result in cubes with many empty bins punctuated by rare hot spots, which hinder the detection of the long-term trends.
- **Large spatial neighborhoods smooth out outliers and group regions with many high-count bins into large hot spots.** Smaller neighborhoods result in slightly better *emerging hot spots count z-score* correlation but provide little information beyond what an event counts cube provides, since no bins beyond the event location and the (small) neighborhood are classed as hot. Conversely, the use of large neighborhoods in event-sparse regions can result in many hot bins with an event count of zero (*i.e.*, the large hot spots around Ndélé in Figure 38 and Figure 39), creating regular "risk patterns" that can be hard to interpret given the irregular administrative boundaries and settlement patterns.

5.2 Discussion

After developing and testing ORÁCULO, we discuss its strengths and shortcomings by comparing its architecture and test results with the state-of-the-art, (reviewed in Chapter 3), and with the processes that MINUSCA uses to maintain situational awareness and their respective constraints, (described in Chapter 2).

5.2.1 Comparison with the State-of-the-Art

To compare ORÁCULO with the state-of-the-art, we begin by discussing its overall architecture. Then, we proceed to compare each main function and its test results with the benchmarks provided by the literature.

The system architecture of ORÁCULO combines event extraction and spatiotemporal analysis in a single pipeline, much like the DSTTM (Farnaghi, Ghaemi, & Mansourian, 2020). However, it also incorporates the analyst supervision step (Event Dataset Supervision) present in Ushahidi (Manning, 2018) and Liveuamap (2020), not only because it ensures that false positive events are removed before they influence the spatiotemporal analysis, but also because human-verified and annotated tweets containing events can be added to the training data, improving system performance.

The design of the entire Event Extractor component revolves around the choice of event frame and data sources. For the event frame, in order to turn event detection into a binary classification problem, we defined a single "conflict-related event" rather than use multiple FrameNet-derived event frames (Li, Cheng, He, Wang, & Jin, 2019), ACE2005 event frames (Liu, Luo, & Huang, 2018) or the seven ACLED event types (Raleigh, Linke, Hegre, & Karlsen, 2010). Thus, we defined a modified 4W event frame (Hamborg, Breitinger, & Gipp, 2019) that divided "Who" into *Agent* and *Target* to capture the offensive/defensive posture of the factions and added the *Effects* argument to measure the severity (casualties) of the event.

As for the chosen data sources, though the choice of scraping the Twitter accounts of relevant news sources (Table 9) for events is counter-intuitive given the small percentage of CAR internet users (International Telecommunication Union, 2020), it is consistent with the literature (Farnaghi, Ghaemi, & Mansourian, 2020) (Avvenuti, Cresci, Del Vigna, Fagni, & Tesconi, 2018) (Liveuamap, 2020) (Raleigh, Linke, Hegre, & Karlsen, 2010), resulted in an unexpectedly large dataset of 8 518 tweets (comparable to the combined 15 825 tweets in the CrisMap datasets), and greatly simplified event extraction, as the Event Extractor did not have to perform joint event extraction across an entire news article like the Giveme5WH1 (Hamborg, Breitinger, & Gipp, 2019), GCNN (Liu, Luo, & Huang, 2018), and LR/MLN (Li, Cheng, He, Wang, & Jin, 2019) event extractors. Nevertheless, we consider that tweets from news sources and the lead paragraph of news articles have significant overlap, so the Event Extractor could be easily modified to extract events directly from news sources' websites by modifying the scraper function to scrape only the leads.

Given the chosen event frame and the format and amount of available training data, we required an easily customizable event extractor/NLU engine that did not require large amounts of training data, so we selected Snips NLU (Coucke, et al., 2018) instead of Giveme5WH1 (Hamborg, Breitinger, & Gipp, 2019). Choosing Snips NLU had the added benefit of reducing the preprocessing pipeline, since most preprocessing tasks and word embedding are performed internally, but the disadvantage that the pretrained word embeddings were likely trained on chatbot-like corpora. Nevertheless, Snips NLU was easily adapted into the event extraction role and to the chosen event frame and arguments, resulting in the snips NLU event extractor function. Its best configurations (Test 8 and Test 11) use oversampling (Kotzé, Senekal, & Daelemans, 2020) and argument gazetteers (Coucke, et al., 2018) to effectively mitigate the low amount of training data and the large class imbalance. Annotation was conducted by two analysts with military and machine learning experience, and training and testing were performed using the annotated *training/validation dataset* and 5-fold cross-validation (80-20 train-test split), which is consistent with the literature (Kotzé, Senekal, & Daelemans, 2020).

Regarding its performance at extracting events and arguments, we can compare the snips NLU event extractor with the following systems:

- The state-of-the-art Giveme5WH1 (Hamborg, Breitinger, & Gipp, 2019), GCNN (Liu, Luo, & Huang, 2018), and LR/MLN (Li, Cheng, He, Wang, & Jin, 2019) event extractors and the Snips NLU engine in its intended purpose (intent resolution) (Coucke, et al., 2018) by assuming that all test procedures are compatible (Table 4);
- The Logistic Regression model used by Kotzé, Senekal, & Daelemans (2020) and the SVM used by CrisMap (Avvenuti, Cresci, Del Vigna, Fagni, & Tesconi, 2018) by applying those models to the ORÁCULO *training/validation dataset* (Table 14).

Versus the state-of-the-art systems, the snips NLU event extractor demonstrates higher recall but lower precision and F1 both on event detection and argument extraction. Better event detection precision can be achieved by increasing the event confidence level threshold, but this also lowers its recall (Table 13), diminishing its ability to extract the largest possible number of conflict-related events from the input tweet stream. Thus, we selected the high recall, high balanced accuracy, low precision Test 8 configuration as the best optimized given the role of ORÁCULO, choosing to rely on the Event Dataset Supervision (analyst supervision) step to remove the false positives. However, test results suggest another approach to improve event detection performance: since the SVM and Logistic Regression classifiers achieved significantly better precision than Snips NLU, SVM/Logistic Regression and Snips NLU could be combined so that the former classified the results of the latter.

Conversely, in argument extraction, though the precision and F1 of the snips NLU event extractor are still below the state-of-the-art, the performance gap is smaller than on event detection, and recall continues to be above par. In part this is likely because the textual patterns of conflict-related news tweets tend to have very little variance, allowing both the deterministic intent parser and CRF-based probabilistic intent parser of Snips NLU (Coucke, et al., 2018) to extract arguments slightly better – assuming test procedures are compatible – than the GCNN (Liu, Luo, & Huang, 2018) and LR/MLN

(Li, Cheng, He, Wang, & Jin, 2019), which has to extract events of various frames with various frame-specific arguments. Oversampling improved argument extraction performance, further suggesting that the monolithic textual patterns help argument extraction, but the other strategies and parameters had little effect. Therefore, improving argument extraction performance even further could require structural changes, such as redefining the "conflict-related event" frame to combine the "Agent" and "Target" arguments into a single "Entities" argument (the "Target" argument proved the most difficult to extract). Another possibility would be to manually train the Snips NLU word embeddings on a corpus of conflict-related events: since the argument extraction performance of our implementation of Snips NLU is below the results achieved during its testing (Coucke, et al., 2018), it is likely that the pretrained Snips NLU word embeddings are performing sub-optimally in the event extraction role.

Geocoding procedures and results were also consistent with the literature. Like Giveme5WH1 (Hamborg, Breitinger, & Gipp, 2019), the geocoder completes the geographic information of the extracted locations and then decides the most specific as that which best describes event location. However, the decision process used by the current implementation of the geocoder is textual rather than truly geographic, as it relies on type codes and administrative levels to decide the most specific location, while Giveme5W1H compares the settlement size (areas and bounding boxes) provided by the geocoder. In part, this was imposed by the context, as the OCHA settlements shapefile (2018) whence the CAR gazetteer came stored settlements as point objects and not as polygons. However, shapefiles for level 1, 2, and 3 administrative regions *do* exist and were loaded into the *event geodatabase* (Figure 25), so the information provided by the geocoder in text format could have been associated to the polygons, and spatial queries could have been used to provide further criteria for the geocoder.loc selector function, partially mitigating the lack of information on settlement size. Nevertheless, the geocoder function still achieved an accuracy of 0.959 which, assuming that the geoparsing F1 of 0.84 achieved by CrisMap (Avvenuti, Cresci, Del Vigna, Fagni, & Tesconi, 2018) is equivalent to its accuracy[39], is above at benchmark. This is likely due to

[39] Geocoding can be thought of as an information retrieval problem: whenever it fails to retrieve the true geographic information of a toponym, it creates a false negative (the geographic information which should have been retrieved) and a false positive (the geographic information it was mistakenly retrieved). Conversely, when it retrieves the correct geographic information, it also creates a true negative (the information which shouldn't have been retrieved and wasn't). Therefore, since F1 is the harmonic mean of precision and recall (Appendix A: Performance Metrics), accuracy, precision, recall, and F1 will be the same when evaluating geocoders if the procedure is the same as described above.

the combined use of the exhaustive CAR gazetteer derived from the OCHA shapefile and the GeoNames web service by callling it with varying degrees of fuzzyness to compensate for transliteration errors.

In contrast, to merge extracted and geocoded events (enriched tweets) into unique events, the event merger relied on spatial and temporal heuristics – time window and similar location, region, or country – instead of using the clustering algorithms used by DSTTM (Farnaghi, Ghaemi, & Mansourian, 2020), which consider the spatial, temporal, and semantic distances between tweets to cluster them into events. Alternatively, if the geocoder stored the *event location* as a geodatabase point object instead of storing the event location, region, and country as textual data, events could have been clustered using a combination of time windows, spatial kernels, and word embeddings (*i.e.*, cosine similarity between vectors of the *preprocessed content* feature). Still, the heuristics approach performed reasonably, achieving an event merging accuracy of 0.812. As for the selection of the most relevant arguments amongst the merged events, only the heuristics to select the unique *event location* and *date* are consistent with the literature, since they are the same used by Gimeve5W1H (Hamborg, Breitinger, & Gipp, 2019): the most specific location and the earliest date. Notably, event merger does not attempt to select the most relevant *action*, *agent*, *target*, and *effects* – it simply stores the pre-merge values in lists and removes the duplicates. Thus, the resulting values can be thought of as event "tags" and can be used to query the *event geodatabase*.

Concerning the assessment of the Event Extractor as a whole, the biggest hurdle was that there is no definite ground truth CAR Civil War event dataset. Therefore, we assumed the *ACLED event dataset* as the best next thing (Duursma, 2017) and measured the amount of mutual information (events) it had with the results of Test 8 (Figure 31). Though there is significant overlap between both datasets, only 23% (77) of the combined number of events are common to both datasets. One possible explanation for this discrepancy is that ACLED does not rely exclusively on online open sources (Raleigh, Linke, Hegre, & Karlsen, 2010), and that their geographically dispersed network of researchers includes *in situ* personnel which can analyze print media and radio. However, this is inconsistent with Figure 15, as more than 90% of the ACLED events during the scraping period were collected from sources with an online presence. Consequently, a more likely explanation is that the overlap is not stronger because ORÁCULO does not extracts events directly from the websites of news sources, while ACLED lacks the resources to continuously monitor Twitter.

Finally, the absence of a ground truth dataset (space-time cube) of CAR spatiotemporal event hot spots/clusters forced the Hot Spots Detector to be developed and tested without any external data-led validation. Such dataset would have allowed the parameters of the emerging hot spots detector to be optimized to the configuration that maximized the similarity of the results with the ground truth. Thus, in the absence of such dataset, we assumed the ACLED and ORÁCULO event datasets as samples of an unknown CAR event "population" with identical distributions, performing parameter sweep across the range of possible emerging hot spots detector parameters to find the configurations which maximized the Pearson correlation coefficients between the ACLED and ORÁCULO *emerging hot spots count z-score* and *trend count z-score*. Doing so revealed a stronger relation (>0.6 Pearson correlation coefficient for the top 10 configurations) (Table 18) between ORÁCULO and ACLED data than the limited overlap between event datasets suggested, supporting the assumption made above, but while the best configurations all had similar temporal bin (1 month) and neighborhood (5 months) sizes, no set of spatial parameters stood out.

Therefore, we visually analyzed the 2- and 3-dimensional visualizations of two best configurations (Figure 34 to Figure 41) to assess the strengths and shortcomings of the competing spatial neighborhood sizes, concluding that smaller spatial neighborhoods add little information beyond that which the event locations themselves provide. Conversely, while the larger neighborhood sizes combine to create meaningful "hot spot surfaces" in regions where event locations are dispersed (*e.g.*, the Northeast), in regions where events are clustered in specific locations (*e.g.*, Ndélé, Bangui), large neighborhood sizes create large regular "hot spots surfaces" where no event ever occurred save for the location at its center, failing to account for the more clustered pattern of the region. This is likely because emerging hot spots analysis is optimized for detecting hot spots in spatially homogenous regions, like a city, where static spatial neighborhoods are meaningful. Consequently, it is likely that using a dynamic spatial neighborhood size – for instance, one dependent on the amount of spatial clustering of settlements within a certain search radius – would have yielded more meaningful results. Another possibility would be to keep the space-time cube data structure but use the OPTICS clustering algorithm (Farnaghi, Ghaemi, & Mansourian, 2020) to discover clusters of high-count space-time bins while taking the spatial heterogeneity of the CAR into account.

5.2.2 Comparison with MINUSCA processes and constraints

The final step section the Discussion compares ORÁCULO with the processes MINUSCA uses to maintain situational awareness and verifies if ORÁCULO addresses some of their constraints. To do so, first, we compare the existing processes and ORÁCULO on intelligence collection (Event Extractor), and then on intelligence processing and analysis (Hot Spots Detector).

Regarding intelligence collection, ORÁCULO partially automates the collection of conflict-related events from online news sources. Thus, ORÁCULO complements rather than competes with the existing intelligence collection processes and tools – including SAGE (Expert Panel on Technology and Innovation in UN Peacekeeping, 2015) –, which focus on collecting HUMINT. Due to the automation of most of its processes, ORÁCULO also does not compete with existing processes for human resources: while it incorporates the analyst supervision step to ensure data quality, supervision is not a full-time job, as briefly sifting through the on average 5 "daily catch" events extracted by ORÁCULO to remove 4 false positives is a far cry from having to manually scrape 22 tweets per day to find only 2 relevant tweets, which also have to be manually annotated, geocoded, merged, and added as events to the event geodatabase – too much manual work for such a meagre, although important, output, especially when considering the small size of the JMAC (Theunens, 2017). Furthermore, as the number of sources and tweets/news/reports per day increases, we can expect that the number of man-hours saved by automating OSINT intelligence collection increases even further.

Where ORÁCULO would compete with existing processes if it were deployed by MINUSCA in its current form would be in the event geodatabase it generates, since it is separate from the one maintained by SAGE. However, it is important to recall that the goal of the present project is the development and testing of a prototype, not the deployment of a fully developed system. If ORÁCULO were to be deployed by MINUSCA, it would first have to be integrated with SAGE. One possible approach to do so given the system architecture of ORÁCULO would be to separate the Event Extractor and the Hot Spots Detector, so that the output of the Event Extractor was merged with the SAGE event dataset before being fed to the Hot Spots Detector in order to detect event hot spots. Thus, the resulting event dataset would consist of events extracted manually from mostly human sources (SAGE) and automatically from online news sources (ORÁCULO). In turn, the expected increase in data (Duursma, 2017) would make the output of the Hot Spots Detector closer to the "true" CAR event hot spots independently of the parameters it used.

The detection of spatiotemporal hot spots of conflict-related activity is the last topic we address. ORÁCULO uses Emerging Hot Spots Analysis (ESRI, 2020e) to discover statistically significant spatiotemporal hot spots of CAR Civil War-related events instead of attempting to cluster events directly. We decided on this analysis technique because it uses spatiotemporal bins as its unit of analysis, addressing the constraint identified by Duursma and Karlsrud (2019): that current processes to maintain the situational awareness of peace operations consider only the event as their basic unit of analysis – a claim that is supported by the limited role which the JMAC Handbook (Martin-Brûlé & Assouli, 2018) ascribes to hot spots maps. Focusing on events instead of areas and time periods can skew peacekeeper decision-making to a game of "whack-a-mole" in which resources are always spent to address the most recent events, while never attempting to distinguish long-standing trouble spots from outliers. In law enforcement, this constraint is addressed by Hot Spots Policing (Braga, Turchan, Papachristos, & Hureau, 2019) (Center for Evidence-Based Crime Policy, 2020), in which the detection of statistically significant spatial hot spots of criminal activity using spatial analysis techniques guides police deployment and intervention. Therefore, since Emerging Hot Spots Analysis is one such analysis technique and has been used to model criminal activity over time in a city-like area (Hashim, Mohd, Sadek, & Dimyati, 2019), in ORÁCULO we attempted to adapt it to model conflict-related activity over time in a country.

It is hard to assess the success of our attempt. Since events and not areas are the current unit of analysis in peace operations like MINUSCA, no ground truth event space-time cube existed to guide the optimization of the Hot Spots Detector. Thus, we devised and implemented an optimization process based on the correlation between space-time cubes generated from two event datasets (ACLED and ORÁCULO) covering the CAR Civil War, and although that process led to possibly meaningful outputs, only a continuous quality assessment by MINUSCA intelligence analysts can permit true optimization.

Chapter Output

In chapter 5, we evaluated the prototype of ORÁCULO by describing how its components were tested separately and how the system was tested a whole, and by discussing the test results in regard to the state-of-the-art of event extraction and spatiotemporal analysis reviewed in chapter 3, and to the constraints to the situational awareness of MINUSCA reviewed in chapter 2. In the next chapter, we use the key findings of this discussion to assess whether the current project goal was achieved, to identify what were the main limitations of this research, and to propose what sort of future work can be conducted to improve and advance the project.

6 Conclusion and Recommendations

Chapter Goals

The goals of Chapter 6 are assessing the achievement of the project goal, identifying the research limitations, and proposing future work on the research problem and on project ORÁCULO. It maps to the “Review Process” and “Determine Next Steps” tasks of the “Evaluation” phase of CRISP-DM and addresses the “Deployment” phase by sketching a deployment plan for ORÁCULO.

6.1 Assessment of Project Goal

Motivated by the research problem of *“How to detect areas and time periods of significant conflict-related activity in peace operations using open-source information?”*, the goal of the present project was *“Developing and testing the prototype of a system, ORÁCULO, capable of detecting spatiotemporal hot spots of conflict-related events extracted from online news sources”*. Therefore, we can split the assessment of the project goal into two parts: the assessment of the ability of ORÁCULO to extract conflict-related events from online news sources and the assessment of its ability to detect spatiotemporal hot spots of events.

Concerning the ability of ORÁCULO to extract conflict-related events from online news sources, we have shown how our choice of test sources – the Twitter accounts of CAR and international news organizations – allowed the extraction of a similar number of events to that found in the ACLED dataset for the same conflict and for the same period. We have also shown how our event extractor based on Snips NLU extracts events better than other state-of-the-art methods given the intelligence collection context, as it achieves better event detection and argument extraction recall. Furthermore, we developed geocoding and event merging functions on par with the state-of-the-art and incorporated analyst supervision in the event extraction process to enforce the quality of the extracted events and to contribute to event extractor training. Nevertheless, despite the inclusion of analyst supervision, the amount of human intervention falls well below that required to perform the equivalent tasks without any event extraction system, addressing an important context-derived constraint. Thus, we can conclude that ORÁCULO can successfully extract events from online news sources.

As to the ability of ORÁCULO to detect spatiotemporal hot spots of conflict-related events, we have adapted emerging hot spots analysis to the context of peace operations, devising a procedure to optimize its parameters using another event dataset. Although the absence of a ground truth hot spots dataset precludes the complete assessment of the quality and reliability of the detected hot spots, the strong correlation between the ORÁCULO and ACLED hot spots space-time cubes suggests that ORÁCULO is indeed successfully detecting statistically significant hot spots of conflict-related events.

Consequently, we can conclude that ORÁCULO can successfully extract events from online news sources and detect their spatiotemporal hot spots, and that the project goal was successfully achieved. Moreover, we can consider the *viability* of the proposed solution to the research problem as proven, although further testing conducted on more use cases (peace operations) is needed to prove its full *effectiveness*.

6.2 Limitations

The results of the research were limited by two types of limitations: data limitations and method limitations. Data limitations were the low amount of effective training and validation data, the low amount of contextual data, and the absence of ground truth data. First, the low amount of training and validation data limited the event extraction performance, as only 965 out of 8 518 tweets could be used as Snips NLU training utterances. Second, the CAR settlements shapefile stored settlements as point objects, preventing us from using their area or population size as decision criteria during the event location selection step of the geocoder and event merger functions. Lastly, the absence of ground truth CAR event datasets and CAR event hot spots limited the full evaluation of ORÁCULO, especially the evaluation of the Hot Spots Detector component.

As for method limitations, they are the unsuitable Snips NLU pretrained word embeddings, the crude event location selection criteria, the crude event merging criteria, and the static spatial neighborhood in the emerging hot spots analysis. First, the Snips NLU pretrained word embeddings were trained on chatbot-like corpora, limiting their performance in the event extraction role. Second, our implementation of geocoding decides the most specific candidate event locations using their frequency and type as decision criteria, limiting its ability to decide between two locations of the same type. Furthermore, it does not store event locations as geographic objects, preventing the use of area-based location selection and distance-based event merging. Third, event merging is achieved by defining a time window and selecting all records within with the same *textual* geographic information (toponym, ADM1, country), limiting the ability of merging events with very close locations, (*e.g.*, a city and its suburbs), but with different toponyms. Lastly, the static nature of the spatial neighborhood parameter of the emerging hot spot analysis limits the ability to take spatial heterogeneity into account, leading to overly large hot spots in regions were events and settlements are highly clustered.

6.3 Future Work

Given the current state of project ORÁCULO, we propose that future work focuses on one medium-term objective and on two long-term goals. The medium-term objective should be the development and testing of a demonstrator. This demonstrator should address the method limitations described above, the low amount of training and validation data, and should attempt to use SVM or Logistic Regression classifiers to classify the results of Snips NLU, improving event extraction precision without lowering recall. It should also incorporate a validation and annotation User Interface and generate an interactive web dashboard as its output, allowing the demonstrator to be fully tested by analysts and decision-makers. Future testing should also be conducted in cooperation with MINUSCA or, if other use cases are tested, with the relevant peacekeeping organizations, allowing the use of SAGE databases or similar resources as ground truth datasets and intelligence staff to evaluate the outputs of ORÁCULO.

Finally, in contrast with the well-defined nature of the medium-term objective, the two long-term goals are more open-ended – and more ambitious. The first goal should be to leverage GCNN or other state-of-the-art algorithms to create a custom conflict-related event extraction system capable of performing joint event extraction, as this would expand the range of suitable sources to include news websites (beyond their lead paragraph) and even print media and radio broadcasts, if the respective optical character recognition and automatic speech recognition components were added. The second goal should be to adapt a non-parametric clustering algorithm, like OPTICS, to detect spatiotemporal hot spots of events in a space-time cube, as this would preserve the spatiotemporal bin as the unit of analysis while diminishing the reliance on static parameters. Together, the continued development and testing of ORÁCULO and the attainment of the two long-term goals should result in a system capable of detecting hot spots of conflict-related activity across many conflicts and with minimal context-specific training, informing decision-makers about the areas which require the largest amount of attention in any given time period. And – recalling Lord Nelson's quote at the beginning –, no one can do very wrong once the real problems are finally found.

ORÁCULO.

References

Academia Militar. (2020, May 04). *Assinatura dos novos Contratos-Programa de Investigação e Desenvolvimento*. Retrieved from Academia Militar: https://academiamilitar.pt/assinatura-dos-novos-contratos-programa-de-investigacao-e-desenvolvimento-2020.html

ACLED. (2019). *Armed Conflict Location and & Event Data Project (ACLED) Codebook*. Retrieved from ACLED Data: https://www.acleddata.com/

Avvenuti, M., Cresci, S., Del Vigna, F., Fagni, T., & Tesconi, M. (2018). CrisMap: a Big Data Crisis Mapping System Based on Damage Detection and Geoparsing. *Information Systems Frontiers*(20), pp. 993-1011. Retrieved from https://doi.org/10.1007/s10796-018-9833-z

Ball, A., & Doumouro, C. (2020). *Snips Natural Language Understanding*. Retrieved from Snips NLU 0.20.2 documentation: https://snips-nlu.readthedocs.io/en/latest/

Berman, E., Felter, J. H., & Shapiro, J. N. (2018). *Small Wars, Big Data*. Princeton, New Jersey, United States of America: Princeton University Press.

Braga, A., Andresen, M., & Lawton, B. (2017). The Law of Crime Concentration at Places: Editors' Introduction. *Journal of Quantitative Criminology*(33), 421-426. doi:10.1007/s10940-017-9342-0

Braga, A., Turchan, B., Papachristos, A., & Hureau, D. (2019). Hot spots policing of small geographic areas effects on crime. *Campbell Systematic Reviews, 15*(3). doi:10.1002/cl2.1046

Brown, M. S. (2015, July 29). *What IT Needs To Know About The Data Mining Process*. Retrieved May 12, 2020, from Forbes: https://www.forbes.com/sites/metabrown/2015/07/29/what-it-needs-to-know-about-the-data-mining-process

Carriere, D. (2020). *Geocoder: Simple, Consistent*. Retrieved from geocoder 1.38.1 documentation: https://geocoder.readthedocs.io

Center for Evidence-Based Crime Policy. (2020). *Hot Spots Policing*. (George Mason University) Retrieved June 06, 2020, from Center for Evidence-Based Crime

Policy: https://cebcp.org/evidence-based-policing/what-works-in-policing/research-evidence-review/hot-spots-policing/

Central Intelligence Agency. (2020, May 20). *Africa: Central African Republic*. Retrieved from CIA World Factbook: https://www.cia.gov/library/publications/the-world-factbook/geos/ct.html

Chainey, S., & Ratcliffe, J. (2005). *GIS and Crime Mapping*. John Wiley & Sons.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). *CRISP-DM 1.0 - A step-by-step data mining guide*. SPSS.

Collins, A. (2016). *Metaphone*. Retrieved from PyPI: https://pypi.org/project/Metaphone/

Coucke, A., Saade, A., Ball, A., Bluche, T., Caulier, A., Leroy, D., . . . Dureau, J. (2018, December 06). *Snips Voice Platform: an embedded Spoken Language Understanding system for private-by-design voice interfaces*. Retrieved from arXiv.org: https://arxiv.org/abs/1805.10190

CS6140 Machine Learning. (2015). *Normalized Mutual Information: Estimating Cluster Quality*. Retrieved from Northeastern University - Khoury College of Computer Sciences: https://course.ccs.neu.edu/cs6140sp15/7_locality_cluster/Assignment-6/NMI.pdf

Dukhan, N. (2017). *Splintered Warfare: Alliances, affiliations, and agendas of armed factions and politico-military groups in the Central African Republic*. Retrieved from Enough Project: https://enoughproject.org/wp-content/uploads/2017/07/TESTSplintered-warfare-LOCKED-7.31.17.pdf

Duursma, A. (2017). Counting Deaths While Keeping Peace: An Assessment of the JMAC's Field Information and Analysis Capacity in Darfur. *International Peacekeeping, 24*, pp. 823-847.

Duursma, A., & Karlsrud, J. (2019). Predictive Peacekeeping: Strengthening Predictive Analysis in UN Peace Operations. *Stability: International Journal of Security & Development, 8*(1). doi:10.5334/sta.663

ESRI. (2020a). *How Emerging Hot Spot Analysis works*. Retrieved June 05, 2020, from ArcGIS Help | Documentation: https://pro.arcgis.com/en/pro-app/tool-reference/space-time-pattern-mining/learnmoreemerging.htm

ESRI. (2020b). *Create Space Time Cube By Aggregating Points*. Retrieved from ArcGIS Pro | Documentation: https://pro.arcgis.com/en/pro-app/latest/tool-reference/space-time-pattern-mining/create-space-time-cube.htm

ESRI. (2020c). *ArcGIS Pro | 2D, 3D & 4D GIS Mapping Software*. Retrieved from ArcGIS Pro: https://www.esri.com/en-us/arcgis/products/arcgis-pro/overview

ESRI. (2020d). *ArcGIS Pro Python Reference*. Retrieved from ArcGIS Pro: https://pro.arcgis.com/en/pro-app/latest/arcpy/main/arcgis-pro-arcpy-reference.htm

ESRI. (2020e). *ArcGIS Pro*. Retrieved from How Kernel Density Works: https://pro.arcgis.com/en/pro-app/latest/tool-reference/spatial-analyst/how-kernel-density-works.htm

ESRI. (2020e). *Emerging Hot Spot Analysis*. Retrieved from ArcGIS Pro: https://pro.arcgis.com/en/pro-app/latest/tool-reference/space-time-pattern-mining/emerginghotspots.htm

ETC. (2020). *Emergency Telecommunications Cluster*. Retrieved June 24, 2020, from Emergency Telecommunications Cluster: https://www.etcluster.org/

Expert Panel on Technology and Innovation in UN Peacekeeping. (2015). *Performance Peacekeeping*. Retrieved June 21, 2020, from https://peacekeeping.un.org/sites/default/files/performance-peacekeeping_expert-panel-on-technology-and-innovation_report_2015.pdf

Farnaghi, M., Ghaemi, Z., & Mansourian, A. (2020). Dynamic Spatio-Temporal Tweet Mining for Event Detection: A Case Study of Hurricane Florence. *International Journal of Disaster Risk Science*(11), pp. 378-393. Retrieved from https://doi.org/10.1007/s13753-020-00280-z

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine, 17*(3), pp. 37-54.

GeoNames. (2020). Retrieved from GeoNames: http://geonames.org/

Getis, A., & Ord, J. K. (1992). The Analysis of Spatial Association by Use of Distance Statistics. *Geographical Analysis, 24*(3), 189-206.

Hamborg, F., Breitinger, C., & Gipp, B. (2019). *Giveme5W1H: A Universal System for Extracting Main Events from News Articles*. Retrieved from aXiv: https://arxiv.org/abs/1909.02766

Hashim, H., Mohd, W., Sadek, E., & Dimyati, K. (2019). Modelling Urban Crime Patterns Spatial Space Time and Regression Analysis. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII-4/W16*, 247-254.

Headquarters, Department of the Army. (2012). *ATP 2-22.9 Open-Source Intelligence*. Washington D.C., United States of America.

Henrique, J. (2020). *GetOldTweets-python*. Retrieved from GitHub: https://github.com/Jefferson-Henrique/GetOldTweets-python

Howard, L. (2019, November 21). *Assessing the Effectiveness of the UN Mission in the Central African Republic*. Retrieved from IPI Global Observatory: https://theglobalobservatory.org/2019/11/assessing-effectiveness-un-mission-central-african-republic/#_ftn1

Hoyer, S., Kleeman, A., & Brevdo, E. (2020). *xarray: N-D labeled arrays and datasets in Python*. Retrieved from xarray 0.16.3 documentation: http://xarray.pydata.org/en/stable/

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering, 9*(3), pp. 90-95. Retrieved from https://ieeexplore.ieee.org/document/4160265

HydroGeoLogic, Inc. (2005, May 04). *Mann-Kendall Analysis for the Fort Ord Site*. Retrieved from Statistics How To: https://www.statisticshowto.com/wp-content/uploads/2016/08/Mann-Kendall-Analysis-1.pdf

Information Management Unit, Deparment of Peace Operations, UN. (2020, April 21). *MINUSCA Mission Fact Sheet*. Retrieved from UN Peacekeeping: https://peacekeeping.un.org/sites/default/files/minusca_apr2020.pdf

International Telecommunication Union. (2020). *Statistics*. Retrieved June 23, 2020, from International Telecommunication Union: https://www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx

Kendall, M. G. (1975). *Rank Correlation Methods* (4th ed.). London: Charles Griffin.

Kotzé, E., Senekal, B. A., & Daelemans, W. (2020). Automatic classification of social media reports on violent incidents in South Africa using machine learning. *South African Journal of Science, 116*(3/4). Retrieved from https://doi.org/10.17159/sajs.2020/6557

Lalmas, M., Kazai, G., Kamps, J., Pehcevski, J., Piwowarski, B., & Robertson, S. (2007). INEX 2006 Evaluation Measures. Comparative Evaluation of XML Information Retrieval Systems. *Fifth Workshop of the INitiative for the Evaluation of XML Retrieval*. Dagstuhl. Retrieved from https://hal.inria.fr/inria-00174121/PDF/inex-2006-metrics.pdf

Li, W., Cheng, D., He, L., Wang, Y., & Jin, X. (2019, Fevereiro 18). Joint Event Extraction Based on Hierarchical Event Schemas From FrameNet. *IEEE Access*. Retrieved Novembro 05, 2019, from https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8643786

Liu, X., Luo, Z., & Huang, H. (2018). Jointly Multiple Events Extraction via Attention-based Graph Information Aggregation. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (pp. 1247-1256). Brussels: Association for Computational Linguistics.

Liveuamap. (2020). *Liveuamap*. Retrieved from https://liveuamap.com/about#history

Longley, P., Goodchild, M., Maguire, D., & Rhind, D. (2005). *Geographical Information Systems and Science* (2nd ed.). Chichester: John Wiley & Sons, Ltd.

Loria, S. (2020). *TextBlob: Simplified Text Processing*. Retrieved from TextBlob 0.16.0 documentation: https://textblob.readthedocs.io/en/dev/

Mann, H. B. (1945). Non-parametric tests against trend. *Econometrica, 13*, 163-171.

Manning, N. (2018, August 8). *Keeping the Peace - The UN Department of Field Service's and Peacekeeping Operations use of Ushahidi*. Retrieved from Ushahidi: https://www.ushahidi.com/blog/2018/08/08/keeping-the-peace-the-un-department-of-field-services-and-peacekeeping-operations-use-of-ushahidi

Martin-Brûlé, S.-M., & Assouli, N. (2018). *Joint Mission Analysis Centre Field Handbook*. UN Department of Peacekeeping Operations.

McGrath, J. J. (2006). *Boots on the ground: Troop Density in Contingency Operations*. Fort Leavenworth, Kansas, United States of America: Combat Studies Institute Press.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013, September 7). *Efficient Estimation of Word Representations in Vector Space*. Retrieved from arXiv: https://arxiv.org/abs/1301.3781

Moran, P. A. (1950). Notes on Continuous Stochastic Phenomena. *Biometrika, 37*(1/2), pp. 17-23. Retrieved March 23, 2021, from www.jstor.org/stable/2332142

Nakaya, T., & Yano, K. (2010). Visualising Crime Clusters in a Space-time Cube: An Exploratory Data-analysis Approach Using Space-time Kernel Density Estimation and Scan Statistics. *Transactions in GIS, 14*(3), pp. 223-239.

National Research Council. (2003). Geospatial Databases and Data Mining. In N. R. Council, *IT Roadmap to a Geospatial Future* (pp. 47-72). Washington D.C., United States of America: National Academies Press.

NATO. (2009). *ATP-3.2.1 Allied Land Tactics*. NATO.

NATO. (2015). *AAP-6(C) NATO Glossary of Terms and Definitions (English and French)*. NATO.

Natural Language Toolkit. (2020). Retrieved from NLTK 3.5 documentation: https://www.nltk.org/

OCHA. (2015). *United Nations Office for the Coordination of Humanitarian Affairs - Annual Report 2014*. United Nations. Retrieved June 19, 2020, from https://www.unocha.org/sites/unocha/files/OCHA_AR_Flipbook_Low2014_0.pdf

OCHA. (2018, August 15). *Central African Republic - Villages and Towns with administrative classification of Central African Republic* . Retrieved from Humanitarian Data Exchange: https://data.humdata.org/dataset/central-african-republic-settlements

Olin, N. (2015). Pathologies of Peacekeeping and Peacebuilding in CAR. In T. Carayannis, & L. Lombard (Eds.), *Understanding the Central African Republic* (pp. 194-218). London: Zed Books.

ORACLE Corporation. (2013). *What is Data Mining?* Retrieved May 12, 2020, from ORACLE Help Center: https://docs.oracle.com/cd/E11882_01/datamine.112/e16808/process.htm#DMCON111

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Perrot, M. (2011). Scikit-learn: Machine Learning in Python. *JMLR, 12*(85), pp. 2825-2830.

Pew Research Center. (2019, July 23). *Digital News Fact Sheet*. Retrieved June 23, 2020, from State of the News Media: https://www.journalism.org/fact-sheet/digital-news/

Pretorius, J., & Matthee, M. (2006). The Impact of Spatial Data on the Knowledge Discovery Process. *Proceedings of the Conference on Information Technology in Tertiary Education, Pretoria, South Africa, 18 – 20 September 2006.* Pretoria. Retrieved May 12, 2020, from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.120.9139&rep=rep1&type=pdf

Raleigh, C., Linke, A., Hegre, H., & Karlsen, J. (2010). Introducing ACLED-Armed Conflict Location and Event Data. *Journal of Peace Research, 47*(5), 651-660.

re — Regular expression operations. (2020). Retrieved from Python 3.9.1 documentation: https://docs.python.org/3/library/re.html

Reinsel, D., Gantz, J., & Rydning, J. (2018). *The Digitization of the World: From Edge to Core*. IDC. Retrieved June 24, 2020, from https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf

Reporters Without Borders. (2020). *Central African Republic*. Retrieved June 24, 2020, from Reporters Without Borders: https://rsf.org/en/central-african-republic

Shimazaki, H., & Shinomoto, S. (2007). A Method for Selecting the Bin Size of a Time Histogram. *Neural Computation*, pp. 1503–1527.

Skipper, S., & Perktold, J. (2010). statsmodels: Econometric and statistical modeling with python. *Proceedings of the 9th Python in Science Conference.*

Smith, S. W. (2015). CAR's History: The Past of a Tense Present. In T. Carayannis, & L. Lombard (Eds.), *Understanding the Central African Republic* (pp. 17-52). London: Zed Books.

Smith, S., & Bruce, C. (2008). *Crimestat III User Workbook.* Washington D.C., United States of America: National Institute of Justice.

SOGEFI Ingénierie Géomatique. (2018). *Ressources*. Retrieved from SOGEFI Ingénierie Géomatique: https://www.sogefi-sig.com/ressources/

Spärck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation, 28*, pp. 11-21.

Strand, H., Rustad, S. A., Urdal, H., & Nygård, H. M. (2019). Trends in Armed Conflict, 1946–2018. *Conflict Trends, 3*.

Sundheim, B. M., & Chinchor, N. A. (1993). Survey of the Message Understanding Conferences. *Proceedings of the workshop on Human Language* (pp. 56-60). United States of America: Association for Computational Linguistics. Retrieved from https://doi.org/10.3115/1075671.1075684

Telep, C. W., & Weisburd, D. (2016). Policing. In D. Weisburd, D. P. Farrington, & C. Gill (Eds.), *What Works in Crime Prevention and Rehabilitation: Lessons from Systematic Reviews* (pp. 137-168). Springer.

The FrameNet Project. (2020). *What is FrameNet?* Retrieved from FrameNet: https://framenet.icsi.berkeley.edu/fndrupal/WhatIsFrameNet

The pandas development team. (2020, February). *pandas-dev/pandas: Pandas*. doi:10.5281/zenodo.3509134

Theunens, R. (2017). Achieving Understanding in Contemporary UN Peace Operations: The Joint Mission Analysis Center. In F. Baudet, E. Braat, J. van Woensel, & A. Wever (Eds.), *Perspectives on Military Intelligence from the First World War to Mali* (pp. 173-198). The Hague, The Netherlands: Springer.

Tse-tung, M. (1965). On Guerrilla Warfare. In *Selected Works of Mao Tse-tung*. Beijing: Foreign Languages Press. Retrieved April 07, 2020, from https://www.marxists.org/reference/archive/mao/works/1937/guerrilla-warfare/index.htm

UN Department of Peacekeeping Operations. (2008). *United Nations Peacekeeping Operations - Principles and Guidelines*. Retrieved June 20, 2020, from https://peacekeeping.un.org/sites/default/files/capstone_eng_0.pdf

United Nations. (2019). *Military Peacekeeping-Intelligence Handbook*.

United Nations. (2020). *MINUSCA | Mission multidimensionnelle intégrée des Nations Unies pour la stabilisation en République centrafricaine*. Retrieved May 19, 2020, from MINUSCA | Mission multidimensionnelle intégrée des Nations Unies pour la stabilisation en République centrafricaine: https://minusca.unmissions.org/

United Nations Department of Political and Peacebuilding Affairs and Centre for Humanitarian Dialogue. (2019, March). *Digital Technologies and Mediation in*

Armed Conflict. Retrieved from UN Peacemaker: https://peacemaker.un.org/sites/peacemaker.un.org/files/DigitalToolkitReport.pdf

United Nations Development Programme. (2015). *Human Development Report 2015 - Work For Human Development*. New York. Retrieved June 19, 2020, from http://hdr.undp.org/sites/default/files/2015_human_development_report_0.pdf

United Nations General Assembly. (2019). *Budget for the United Nations Multidimensional Integrated Stabilization Mission in the Central African Republic for the period from 1 July 2019 to 30 June 2020*.

United Nations Geospatial Information Section. (2020, June). *MINUSCA - June 2020*. Retrieved June 19, 2020, from United Nations Geospatial Information Section Website: https://www.un.org/Depts/Cartographic/english/htmain.htm

United Nations Security Council. (2014, April 10). *Resolution 2149 (2014)*. Retrieved from United Nations Digital Library: https://digitallibrary.un.org/record/768393

United Nations Security Council. (2019a, February 15). *Political Agreement for Peace and Reconciliation in the Central African Republic*. Retrieved from UN docs: https://undocs.org/en/S/2019/145

United Nations Security Council. (2019b, November 15). *Resolution 2499 (2019)*. Retrieved from United Nations Digital Library: https://digitallibrary.un.org/record/3836087

United Nations Security Council. (2020, February 14). *Central African Republic - Report of the Secretary General*. Retrieved from MINUSCA: https://minusca.unmissions.org/sites/default/files/sg_report_on_central_african_republic_14_february_2020_en.pdf

Vinck, P., Pham, P. N., Balthazard, M., & Magbe, A. (2019). *Peace, Justice and Security Polls, Report 4*. Harvard Humanitarian Initiative, United Nations Developement Program.

Walker, C., Strassel, S., Medero, J., & Maeda, K. (2006, February 15). *ACE 2005 Multilingual Training Corpus*. doi:https://doi.org/10.35111/mwxc-vh88

Wei, X., & Bang, W. (2019). A Survey of Event Extraction from Text. *IEEE Access*, p. https://www.researchgate.net/publication/337638438_A_Survey_of_Event_Extraction_From_Text.

Wikipedia. (2021a). *Evaluation of binary classifiers*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Evaluation_of_binary_classifiers

Wikipedia. (2021b). *F-score*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/F-score

Wikipedia. (2021c). *Cohen's Kappa*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Cohen%27s_kappa

Wikipedia. (2021d). *Receiver operating characteristic*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Receiver_operating_characteristic

Wikipedia. (2021e). *Entropy (information theory)*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Entropy_(information_theory)

Wikipedia. (2021f). *Mutual Information*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Mutual_information

Wikipedia. (2021g). *Pearson correlation coefficient*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Pearson_correlation_coefficient

Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a Standard Process Model for Data Mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*. Manchester.

Yong, G., Jing, C., Haohan, M., & Yu, L. (2019). Measuring spatio-temporal autocorrelation in timeseries data of collective human mobility. *Geo-spatial Information Science, 22*(3), pp. 166-173. doi:10.1080/10095020.2019.1643609

Yuan, M., Buttenfield, B., Gahegan, M. N., & Miller, H. (2004). Geospatial Data Mining and Knowledge Discovery. In R. McMaster, & L. Usery (Eds.), *Research Challenges in Geographic Information Science*. John Wiley & Sons.

Левенштейн, В. И. (1965). Двоичные коды с исправлением выпадений, вставок и замещений символов. *Докл. АН СССР, 163*(4), pp. 845-848. Retrieved from http://www.mathnet.ru/php/archive.phtml?wshow=paper&jrnid=dan&paperid=31411&option_lang=rus

Appendix A: Performance Metrics

Recall, or True Positive Rate

$$R_y = \frac{TP}{TP + FN}$$

The recall *R* of a binary classifier for a given class y is given by the equation above, where:

> *TP* represents the number of features of class y correctly classified as y (true positives);
> *FN* represents the number of features of class y incorrectly classified as non-y (false negatives);

Source: (Wikipedia, 2021a).

Precision

$$P_y = \frac{TP}{TP + FP}$$

The precision *P* of a binary classifier for a given class y is given by the equation above, where:

> *TP* represents the number of features of class y correctly classified as y;
> *FP* represents the number of features of class non-y incorrectly classified as y;

Source: (Wikipedia, 2021a).

Specificity, or True Negative Rate

$$TNR_y = \frac{TN}{TN + FP}$$

The true negative rate *TNR* of a binary classifier for a given class y is given by the equation above, where:

> *TN* represents the number of features of class non-y correctly classified as non-y;
> *FP* represents the number of features of class non-y incorrectly classified as y;

Source: (Wikipedia, 2021a).

Accuracy

$$Acc_y = \frac{TP + TN}{TP + TN + FP + FN}$$

The accuracy *Acc* of a binary classifier for a given class y is given by the equation above, where:

> *TP* represents the number of features of class y correctly classified as y (true positives);
> *TN* represents the number of features of class non-y correctly classified as non-y;
> *FN* represents the number of features of class y incorrectly classified as non-y (false negatives);
> *FP* represents the number of features of class non-y incorrectly classified as y;

Source: (Wikipedia, 2021a).

Balanced Accuracy

$$BA_y = \frac{R_y + TNR_y}{2}$$

The balanced accuracy *BA* of a binary classifier for a given class y is given by the equation above, where:

> R_y represents the recall or true positive rate of the binary classifier for class y;
> TNR_y represents the true negative rate of the binary classifier for class y;

Source: (Wikipedia, 2021a).

F-score

$$F_{\beta y} = \frac{(1+\beta^2) \times R_y \times P_y}{(P_y \times \beta^2) + R_y}$$

The F-score F_β of a binary classifier for a given class y is given by the equation above, where:

> R_y represents the recall of the binary classifier for class y;
> P_y represents the precision of the binary classifier for class y;
> β is a real number represents the number of times that recall is considered more important than precision.

The popular F1 score is simply a F-score with a β of 1.

Source: (Wikipedia, 2021b).

Mean Average Generalized Precision (MAgP)

$$MAgP = \frac{\sum_{i=1}^{Q}(\frac{\sum_{j=i}^{r} F(a_j)}{r})}{Q}$$

The Mean Average Generalized Precision *MAgP* of an information retrieval system which answers each query with a ranked list of results is given by the equation above, where:

> Q represents the number of queries;
> r represents the number of ranks in the results;
> a represents each ranked result;
> $F(a_j)$ represents the F1 of each ranked result (predicted rank *versus* true rank).

Source: (Lalmas, et al., 2007).

Cohen's Kappa

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

The Cohen's Kappa between two classifiers is given by the equation above, where:

> p_o represents the observed agreement between both classifiers, *i.e.*, ratio between the number of records classified equally by both classifiers and the total number of records, *i.e.*, it is identical to *Acc*;
> p_o represents the probability that the agreement between classifiers is the product of chance.

Source: (Wikipedia, 2021c).

False Positive Rate

$$FPR_y = \frac{FP}{TN + FP}$$

The false positive rate *FPR* of a binary classifier for a given class y is given by the equation above, where:

> *TN* represents the number of features of class non-y correctly classified as non-y;
> *FP* represents the number of features of class non-y incorrectly classified as y;

Source: (Wikipedia, 2021a).

Area Under the Receiver Operating Characteristics curve (AUROC)

$$AUROC = \int_{x=0}^{1} R(FPR(T))dx$$

The Area Under the Receiver Operating Characteristics curve *AUROC* of a classifier is given by the equation above, where:

> *R* represents the recall of the classifier given its False Positive Rate *FPR*;
> *FPR* represents the False Positive Rate of the classifier given a performance threshold *T*, which is normalized between 1 and 0;

Source: (Wikipedia, 2021d).

Entropy

$$H(X) = -\sum_{x \in X} P(x_i) \log_b P(x_i)$$

The Entropy *H* of a discrete variable *X* is given by the equation above, where:

> *x* represents a possible value of *X*;
> *P(x)* represents the probability of *X* having the value *x*;
> *b* is the base of the logarithm used. When *b* = 2, *H* is measured in “bits”.

Source: (Wikipedia, 2021e).

Mutual Information

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p_{(X,Y)}\,(x,y) \log_b (\frac{p_{(X,Y)}(x,y)}{p_{(X)}(X) p_{(Y)}(Y)})$$

The Mutual Information *I* of two discrete variables *X* and *Y* is given by the equation above, where:

> *x* and y represent possible values of *X* and *Y*;
> $p_{(X)}(x)$ represents the probability of *X* having the value *x*;
> $p_{(Y)}(y)$ represents the probability of Y having the value y;
> $p_{(X,Y)}(x,y)$ represents the probability of *X* having the value *x* and *Y* having the value y;
> *b* is the base of the logarithm used.

Source: (Wikipedia, 2021f).

Normalized Mutual Information (NMI)

$$NMI(X;Y) = \frac{2 \times I(X;Y)}{H(X) + H(Y)}$$

The Normalized Mutual Information *NMI* between datasets *X* and *Y* is given by the equation above, where:

I(X;Y) represents the Mutual Information between *X* and *Y*;
H(X) represents the Entropy of *X*;
H(Y) represents the Entropy of *Y*;
Source: (CS6140 Machine Learning, 2015).

Pearson Correlation Coefficient

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

The sample Pearson Correlation Coefficient *r* between variables *X* and *Y* is given by the equation above, where:

n represents the sample size;
x_i represents the values of *x* for sample point *i*;
y_i represents the values of y for sample point *i*;
$\bar{x}$, $\bar{y}$ represent the sample means of *x* and *y*;
Source: (Wikipedia, 2021g).

C&SIG

UNIGIS PT

NOVA
IMS
Information
Management
School