

Extracting Victim Counts from Text

Mian Zhong Shehzaad Dhuliawala Niklas Stoehr

Institute for Machine Learning, ETH Zürich

mzhong@ethz.ch shehzaad.dhuliawala@inf.ethz.ch niklas.stoehr@inf.ethz.ch

Abstract

Decision-makers in the humanitarian sector rely on timely and exact information during crisis events. Knowing how many civilians were injured during an earthquake is vital to allocate aids properly. Information about such *victim counts* is often only available within full-text event descriptions from newspapers and other reports. Extracting numbers from text is challenging: numbers have different formats and may require numeric reasoning. This renders purely string matching-based approaches insufficient. As a consequence, fine-grained counts of injured, displaced, or abused victims beyond fatalities are often not extracted and remain unseen. We cast victim count extraction as a question answering (QA) task with a regression or classification objective. We compare regex, dependency parsing, semantic role labeling-based approaches, and advanced text-to-text models. Beyond model accuracy, we analyze extraction reliability and robustness which are key for this sensitive task. In particular, we discuss model calibration and investigate few-shot and out-of-distribution performance. Ultimately, we make a comprehensive recommendation on which model to select for different desiderata and data domains. Our work is among the first to apply numeracy-focused large language models in a real-world use case with a positive impact.¹

1 Introduction

Timely and accurate information during crisis events is crucial for rescue operations and the allocation of humanitarian aid (Lepuschitz and Stoehr, 2021). However, crisis information is often scarce, subjective, or biased, which renders reported numbers in text extremely important (Hellmeier et al., 2018; Zavarella et al., 2020; Radford, 2021). For instance, the count of injured or missing people provides quantitative information about the catastrophic impact of an earthquake. In this work, we

focus on human victims in crisis events, e.g., fatalities in floods, herein referred to as *victim counts*. A reliable estimate of victim counts is helpful during crisis (Darcy and Hofmann, 2003; Kreutzer et al., 2020), and also post-crisis, benefiting research to diversify measures of crisis intensity. As of now, most intensity measures are either limited to event types (Vincent, 1979; Goldstein, 1992), fatality counts (Kalyvas, 2006; Chaudoin et al., 2017) or both (Stoehr et al., 2022). More fine-grained measures such as injured, displaced, or abused victims are not captured in most popular databases and remain unmonitored (Krause, 2013; Cruyff et al., 2017; Cullen et al., 2021).

Many victim counts are reported in full-text form within event descriptions in news media. This makes their systematic collection and analysis technically complex. Manual extraction of victim counts from text is very labor-intensive and does not scale to big data collections (Schrodt and Ulfelder, 2016; Lewis et al., 2016). Computerized approaches such as the event coding software Tabari (Schrodt, 2009) and Petrarch2 (Norris et al., 2017) focus on extracting actor and event types. They rely on lambda calculus and syntactic pattern matching, but disregard mentions of victim counts.

As we will show, parsing-based approaches perform decently well at extracting explicitly reported victim counts. They can identify the mention of the count “5” in “5 people were injured”. However, they are often inadequate when the description *implies* a correct count — for example, from the description that “one logger was shot but survived”, a human reader may infer that *one* person is injured. Since neither a count nor the injury is mentioned explicitly, a parsing-based system may fall short. Another difficulty stems from the fact that the counts can be reported in many, different formats. A reported count may be digit-based or spelled out, define an exact quantity or a range as in “dozens of people were injured”. As a conse-

¹Code is available online at:
https://github.com/mianzg/victim_counts

quence, formulating the task of victim count extraction is not an easy endeavor (§3). Most prior work assumes a setting where the count is explicitly mentioned in an event description (Döhling and Leser, 2011; Imran et al., 2013; Rudra et al., 2018; Camilleri et al., 2019). Such settings can be tackled by sequence labeling models that select a relevant span from the given description. However, if the victim count does not appear verbatim, as in the above “one logger” example, models with some form of abstract reasoning capacity may be needed (Roy et al., 2015). Recently, large language models have shown promising results in answering number-focused questions with and without explicit mentions of relevant numbers (Lewkowycz et al., 2022; Nye et al., 2021; Wei et al., 2022; Lefebvre and Stoehr, 2022).

This paper is concerned with studying these different approaches (§4): as baselines, we compare regular expression, dependency parsing, and semantic role labeling. We consider the NT5 (Yang et al., 2021) model as a representative numeracy-enhanced pre-trained language model. We use the representation of this model in a generation, a classification, and a regression setting. We evaluate all models along three dimensions: accuracy (§5), reliability (§6), and robustness (§7). We find that the fine-tuned language model outperforms the baseline models, especially when the victim count extraction requires reasoning. Reliability and robustness are particularly important in high-stake, human-centric tasks such as victim count extraction (Zhang et al., 2020; Kong et al., 2020; Russo et al., 2022b). Model reliability indicates to which extent model behavior can be trusted within decision-making settings (Leibig et al., 2017; Jiang et al., 2021). One dimension of reliability is model calibration which indicates if a model’s confidence is aligned well with it making correct predictions (Guo et al., 2017). While calibration has been widely studied for classification, we add to the discussion of calibrated regression (Song et al., 2019) and generation settings (Widmann et al., 2021). Finally, the dimension of robustness describes how stably a model performs. For instance, when the training set is limited or when the test data is out-of-distribution, a less robust model will forfeit more of its predictive performance. To shed light on this dimension, we conduct experiments in few-shot learning and out-of-distribution settings.

We conclude with an application to showcase the extraction of fine-grained and highly specialized types of victim counts. Lastly, we discuss the benefits and drawbacks of the different approaches to assist practitioners in choosing the most suitable task formulation and model.

2 Data

We use publicly available datasets covering natural disasters and armed conflicts, namely: (1) **World Atrocities Dataset (WAD)** (Schrodt and Ulfelder, 2016), (2) **Non-violent and Violent Campaigns and Outcomes 3.0 (NAVCO)** (Lewis et al., 2016), and (3) **European Media Monitor (EMM)** (JRC Science Hub, 2018; Steinberger et al., 2017). For each dataset, we use the event text description and two types of victim counts: the death count and the injury count that we refer to as “WAD death” or “WAD injury”. We pre-process the data by removing the samples with missing values (NaN) in the victim counts . For EMM, we only consider samples with a non-zero victim count since “0” is over-represented.

3 Task Formulation

In this section, we discuss some questions and challenges faced in formulating the task of extracting victim counts from event descriptions. We justify some of the choices we make and describe why it is not possible to have a single formulation that fits all needs:

Is the victim count always present in the text?

Victim counts can be expressed in various ways in the text. When the count is expressed explicitly in the text, say “5 people were injured”, a span extraction model can effectively extract the injury count 5. However, in certain cases, a single explicit number might not be mentioned, and the victim count needs to be logically or algebraically inferred from the text. Consider the description “a 4-year-old girl and her mother were found dead”; a model would need to logically deduce that the victim count of death is 2. To handle this, we not only look at span extraction models but also experiment with models that can understand the text at a deeper level and produce a victim count.

Is the victim count always a single number?

Often, in the event description, the victim count is described as a range, such as “at least 330 people died”, or in vague terms, like “dozens were

injured”. Additionally, even within a description, the victim counts for the same event can be varying, possibly because of recording the counts from different sources. This makes extracting a single exact count almost impossible. In such cases, the best a model can do is to output a close estimate of the actual victim count. Another solution would be to provide a range within which the count could lie. For a humanitarian section deciding on the quantity of aid to be deployed, a range might suffice over a single exact count. To account for this, we also look at models that are trained to output a range by classifying the victim counts into a set of binned categories.

4 Models

In §4.1, we introduce baselines models that parse an event description and heuristically extract a victim count. We then specify the model implementation for the different task formulations in §4.2.

4.1 Baseline Models

All baselines extract a victim count by locating the part of the text that could be relevant to victims and finding the nearby victim counts. The locating step requires a pre-defined list of words denoted as *locating list*. For example, to extract death counts, this list would include terms like “kill” and “die”.

Regex. Regular expressions (regex) is a rule-based method to extract counts by string pattern matching. The patterns (App. A) are built based on active or passive voice to extract a count closest to phrases in the locating list.

Dependency Parsing. The dependency parsing model collects all possible numeric modifiers and their dependency relationships. Since not every numeric modifier relates to victim counts, e.g., “42-year-old”, we construct dependency rules with the locating list to decide if the number is the victim count. For example, one rule is to check if the numeric modifier is for a subject phrase that would reject “42” in the example of “42-year-old”. If no numeric modifier is found (e.g., “a journalist was injured”), additional rules use the locating list to return “1” if the rule is satisfied and otherwise return “0”.

SRL. Semantic role labeling (SRL) recursively decomposes text input into pairs of predicates and their arguments. We define a list of predicate verbs for death and injury count as the locating list. Then,

we iterate over the predicate-argument pairs, check if any predicate from the locating list occurs, and extract the count from its argument if possible. If a predicate exists, the implementation returns the first number as the count if multiple are found and returns “1” if no verbatim number is found. If no such predicate appears, the count is set to “0”.

4.2 Task Modeling

We perform victim count extraction using three methods: generation, regression, and classification. As discussed above, each of these approaches caters to the different formulations of our task and can be beneficial in different scenarios. Across these methods, we use the same underlying NT5 model. For clarity, we denote NT5-Gen, NT5-Reg, and NT5-Clf for the corresponding models. The NT5 model (Yang et al., 2021) is a variant of the T5 model (Raffel et al., 2020) with further fine-tuning on numerical tasks. We query the model in a similar fashion to previous works by giving the question and event description in the form: “answer me:[question] context:[passage]”. We discuss how we fine-tune this model for each of our specific methods below.

Generation. For generation, we fine-tune NT5 to decode the victim counts auto-regressively. At inference, we use beam search to generate output. Generation does not guarantee to only generate numeral tokens; therefore, we follow De Cao et al. (2021) to constrain the possible generation tokens in a prefix-conditioned way, such that only number digit tokens 0 – 9 and EOS token are allowed at each decoding step.

Regression. For regression, we add two linear layers (with ReLU activation) on the encoder representation to output the numerical victim count. The model is trained to optimize the log mean-squared error between the true and predicted count.

Classification. We model the task as a classification problem by binning the victim counts into ordinal classes. Similar to regression, the model has a classification head of a linear layer and a softmax layer on top of an encoder initialized with NT5 weights. Our experiments use a 3-class classification by converting the victim counts into three categories: [0, 3], (3, 10], (10, ∞).

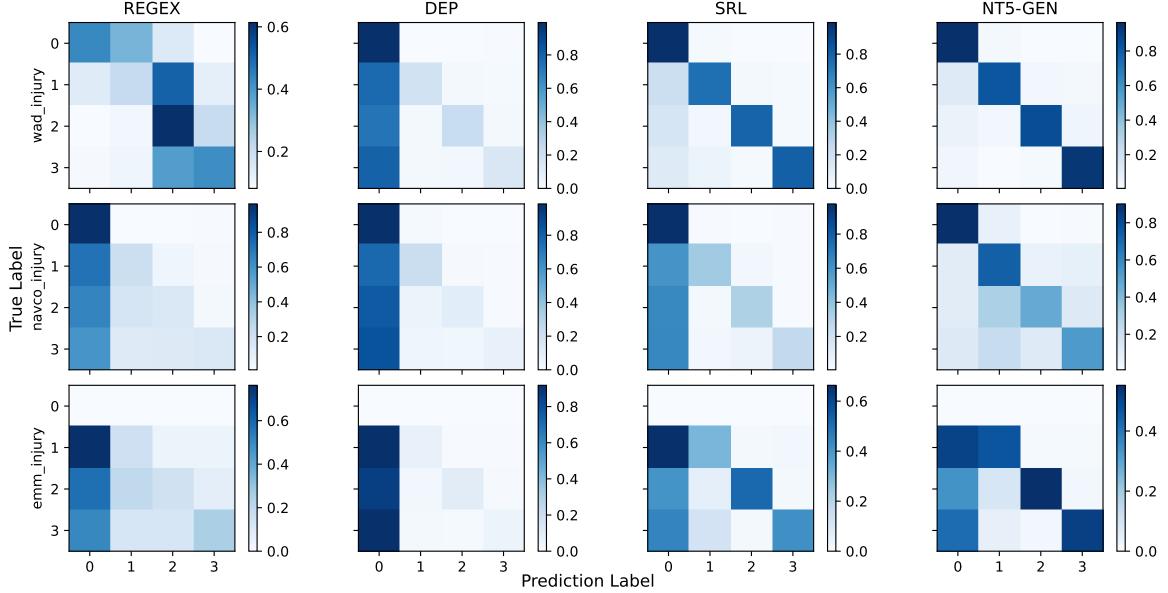


Figure 1: Confusion matrices of the baselines and the fine-tuned NT5-Gen model (columns) of extracting injury counts from different data (rows). We convert the true and prediction victim counts into 4 categories: for any count y , “0” is $y = 0$, “1” is $0 < y \leq 3$, “2” is $3 < y \leq 10$ and “3” is $y > 10$. Values are normalized over true counts. Baselines tend to have low precision on extracting injury counts (dark columns on “0”). SRL and NT5-Gen have comparable accuracy and recall; however, NT5-Gen is slightly better in precision.

5 Accuracy of Counts Extraction

We begin by evaluating the efficacy of our proposed methods for victim count extraction. We examine the model accuracy by comparing baselines and the fine-tuned model with a generation objective (§5.1). We then show the results of using classification and regression formulations (§5.2).

	Exact-Match			F_1		
	WAD	NAVCO	EMM	WAD	NAVCO	EMM
Regex	0.117	0.264	0.064	0.202	0.318	0.124
Dep	0.226	0.303	0.052	0.355	0.363	0.136
SRL	0.741	0.430	0.313	0.779	0.484	0.361
NT5-Gen	0.813	0.501	0.443	0.846	0.544	0.492

Table 1: *Exact-Match* and F_1 scores of the baseline models and the fine-tuned NT5-Gen on injury counts. The best results are **bolded**. The NT5-Gen model performs better than baselines across all datasets. DEP refers to the dependency parsing model and SRL refers to the semantic role labeling model.

5.1 Comparing Baselines with NT5-Gen

We compare the accuracy performance of the baseline models and the fine-tuned NT5-Gen model. Tab. 1 shows the results of extracting the injury counts using *Exact-Match* and F_1 scores commonly used in related tasks (Yang et al., 2021; Dua et al.,

2019). We measure F_1 score on digitized tokens (i.e., “34” → [“3”, “4”]). The fine-tuned NT5-Gen model has an accuracy boost up by 7-13% in *Exact-Match* and by 6-13% in F_1 score than the strongest baseline model SRL. The performance of regex and dependency parsing varies heavily across different data, which implies that the regex pattern or dependency relationship may be less helpful in finding the victim counts.

Moreover, we convert the victim counts into four bins, where the bins are selected to have a balanced number of samples in each bin. As an illustration, Fig. 1 shows the confusion matrices on the transformed injury counts. For both victim types, baseline models have a low precision to falsely return “0” too often. Compared with baselines, the NT5-Gen model improves to extract victim counts whose numeric values are large (e.g., $y > 10$).

Qualitative Analysis. We qualitatively examine error samples of the SRL model that the NT5-Gen model extracts correctly. We randomly select 20 error samples for each test set to evaluate and summarize 4 types of errors with examples in Tab. 2. Out of all errors², 39.2% belong to diverse lin-

²There are a few samples where the ground truth might be erroneous. As the event-coding requires more domain expertise within the corresponding social science discipline, we leave the discussion out of this work.

Error Type	Context	Truth	SRL	NT5
Diverse Expression	<i>Six passengers</i> in a taxi also <i>had their throats cut</i>	6	0	6
Numerical Reasoning	Herders shot and <i>killed four people</i> [...]. Herders then shot and <i>killed a farmer</i> at Jokhana [...]	5	4	5
Number Ambiguity	<i>Unidentified gunmen</i> clash with army	1	0	1
Number Spelling	.Twenty-three people were killed [...]	23	1	23

Table 2: Error examples of SRL that the NT5-Gen model is correct on extracting death counts. Diverse Expression refers to the string patterns not captured by pre-defined rules. Numerical Reasoning shows that the correct count has to be achieved by some mathematical operation over the text. Number Ambiguity indicates that a verbatim number is not written but an estimate may be made (with domain expertise). Number Spelling refers to problems with number / text format that are typos or the tokenizer parses wrongly (e.g., “twenty-three” → “twenty”).

guistic expressions on depicting victims, 38.3% contain number ambiguity, 8.3% need numerical reasoning, and 5.8% have spelling issues (for the tokenizer). The NT5-Gen model performs better when the count needs numerical reasoning. Even if the reasoning is not needed, SRL may fail when the linguistic expression to depict victims (e.g., “have throats cut”) is out of the pre-defined locating list (e.g., [“die”, “kill”, “slay”]). These error types are difficult for baseline models to be improved since the patterns cannot be defined beforehand.

5.2 Results on Classification and Regression

We examine the accuracy of the classification and regression formulations by comparing NT5-Clf and NT5-Reg with different initialization weights. To compare, we use T5-SMALL and BERT-BASE-UNCASED pre-trained weights for the encoder. Tab. 3 shows the classification results on NAVCO injury data. Fine-tuning T5-SMALL and NT5 reaches comparable performance; precision and recall scores are similar, but precision is slightly higher. The scatter plots (Fig. 2) show the results of regression using different pre-trained weights with the mean squared error (MSE). For a (log-transformed) victim count larger than 5, using the regression objective seems more conservative in giving small-valued predictions. The numeracy-rich NT5 weights do not particularly improve accuracy for a classification or regression objective, and employing some standard pre-trained weights might be sufficient.

6 Evaluating Reliability

Another important dimension is reliability which we evaluate through the lens of calibration (§6.1). As we approach the task with multiple formulations, calibration analysis is especially needed to

	Accuracy	F ₁	Precision	Recall
NT5	0.65	0.60	0.62	0.59
T5	0.65	0.60	0.61	0.59
BERT	0.52	0.23	0.17	0.33

Table 3: Classification results on NAVCO injury data with the NT5-Clf model initialized by different pre-trained weights: NT5, T5-SMALL, and BERT-BASE-UNCASED. F₁, precision and recall scores are macro.

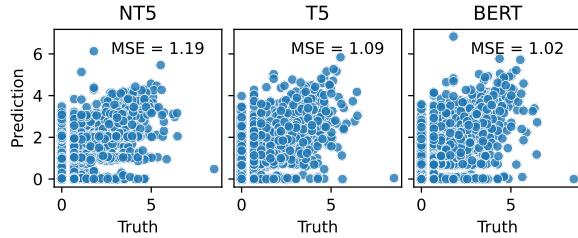


Figure 2: Scatter plots of the fine-tuned NT5-Reg model initialized with different pre-trained weights (NT5, T5-SMALL, and BERT-BASE-UNCASED). The models are trained on log-transformed victim counts.

understand whether a model is calibrated (§6.2), and how post-hoc calibration techniques may adjust models to be better calibrated (§6.3).

6.1 Preliminaries: Calibration Metrics

A well-calibrated model ensures that the confidence of the output is well aligned with the chance of the output being accurate. This is a desirable property for our task — consider a model extracts “0” when the text depicts an injured person. A calibrated model would assign very low confidence to the extracted count, which may avoid error propagation to downstream decisions, e.g., medical resource dispatch. We here introduce the expected calibration error (ECE) (Pakdaman Naeini et al., 2015), a standard metric used for classification and is ex-

tended for generation decoding (Widmann et al., 2021). For regression, we apply quantile calibration error (Kuleshov et al., 2018).

Given n samples, we create M equal-width bins over the interval $[0, 1]$. ECE takes a weighted average on the differences between the classification accuracy and the mean confidence within each B_m ,

$$\text{ECE} = \sum_{m=1}^M \frac{|B_m|}{n} \left| \text{acc}(B_m) - \text{conf}(B_m) \right|.$$

The quantile calibration error averages the differences between the empirical frequency $\text{freq}(B_m)$ and the upper bound of B_m (i.e., $\text{sup}(B_m)$), where $\text{freq}(B_m)$ is the fraction of n samples whose quantiles lower or equal to $\text{sup}(B_m)$,

$$\text{RegCE} = \frac{1}{M} \sum_{m=1}^M \left| \text{freq}(B_m) - \text{sup}(B_m) \right|.$$

The calibration error of generation decoding takes the best b beam search answers, and applies softmax on their scores to represent the confidence. The ECE is then calculated on the best beam search answer similar to classification.

6.2 Calibration Error on Different Models

Data	Model	Death		Injury	
		Orig	Calib.	Orig	Calib.
NAVCO	Clf	0.222	0.044	0.332	0.060
	Reg	0.220	0.097	0.141	0.057
	Gen	0.054	0.040	0.092	0.092
WAD	Clf	0.192	0.055	0.228	0.088
	Reg	0.272	0.107	0.167	0.294
	Gen	0.218	0.221	0.096	0.042
EMM	Clf	0.277	0.098	0.314	0.055
	Reg	0.201	0.189	0.368	0.188
	Gen	0.087	0.092	0.328	0.122

Table 4: Calibration errors of fine-tuned NT5-Clf, NT5-Reg, and NT5-Gen models before (Orig.) and after (Calib.) applying post-hoc calibration. Post-hoc calibration effectively reduces the errors.

We show in Tab. 4 the calibration errors measured on the fine-tuned NT5-Clf, NT5-Reg, and NT5-Gen with different datasets. Surprisingly, the NT5-Gen model is well-calibrated on most datasets, except for EMM injury: the lowest calibration error is 0.05 on NAVCO death, and the errors on

other data range between 0.08 and 0.33. Classification models tend to have large calibration errors (> 0.19). In particular, the error is larger than 0.3 on NAVCO and EMM data to classify injury counts. Regression is also prone to large calibration errors (> 0.15).

Another helpful tool is the reliability diagrams which visualize the calibration errors at different confidence bins. As an illustration, Fig. 3 shows the diagram of the NT5-Clf model fine-tuned on NAVCO injury data, and the diagonal line indicates the perfect calibration. This model is over-confident, and we can observe large gaps when the model confidence is larger than 0.8.

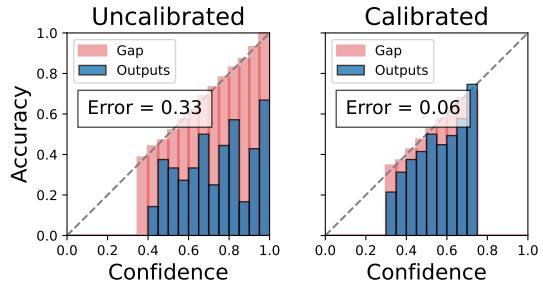


Figure 3: Reliability diagrams compare the calibration error before (left) and after (right) post-hoc calibration of the fine-tuned NT5-Clf model using the NAVCO injury data. This model is prone to large calibration errors (red gaps) in many bins. This is especially true for bins with high model confidence (> 0.8).

6.3 Post-hoc Calibration

Since the models can be over-confident based on the above analysis, we see the necessity to calibrate models for victim count extraction. We use temperature scaling for classification and generation decoding, and isotonic regression for regression. The post-hoc calibrators use development data to minimize negative log-likelihood and are then applied to test sets to measure calibration errors. As a comparison, Fig. 3 (right) shows the calibrated results of the fine-tuned NT5-Clf model on NAVCO injury data. The calibration error (i.e., ECE) reduces from 0.33 to 0.06. The errors of other calibrated models can be found in Tab. 4. In general, when the models have rather a large calibration error (e.g., > 0.3), post-hoc calibration is more helpful and adjusts the models to a better-calibrated level.

7 Evaluating Robustness

Typically, conflict or disaster data is noisy and limited. This is making it challenging to train models

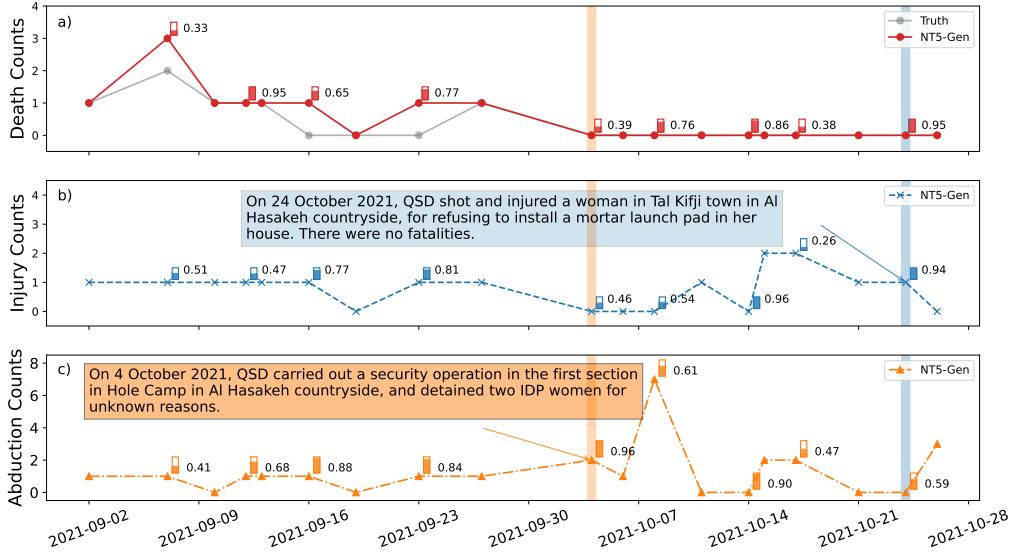


Figure 4: Timeline of victim counts in Syria data from Sept to Nov 2021 as given in the ACLED dataset. We use the NT5-Gen model that is fine-tuned on NAVCO data. Our model can be tested on the extraction of **fatality counts** which is the only victim count featured in ACLED (Fig. (a)). Beyond fatality counts, it can extract more fine-grained victim types such as (b) **injury** and (c) **abduction** counts. Confidence scores are shown for some of the predictions.

on a large-scale, high-quality training set. For this reason, we need robust models that excel in few-shot and out-of-distribution settings.

Reduced Training Size. We fine-tune the NT5-Gen, NT5-Reg, and NT5-Clf models on different-size portions of the training set. Specifically, we use 100%, 50%, 10%, 5%, 0.5% and 0% of the training data and as further discussed in App. C.1. As expected, we find that the accuracy of all models drops when using a smaller training set. The NT5-Gen model reveals to be the most robust in keeping the *Exact-Match* metric above 0.6 when being fine-tuned on only 5% of the training data. The calibration error of the fine-tuned NT5-Clf model increases when the training size is reduced, while the fine-tuned NT5-Reg and NT5-Gen models do not follow this trend. In the zero-shot setting, the NT5-Reg and NT5-Gen models reach their largest calibration error. In contrast, the NT5-Clf model reaches its smallest calibration error in the zero-shot setting.

Out-of-distribution (OOD) Setting. We set up synthetic tasks in which a fine-tuned model is confronted with an out-of-distribution setting at test time. For example, we fine-tune a model on WAD death and then repurpose it to classify WAD injury. Then, we evaluate the drop in performance

of this “out-of-distribution” model compared to an “in-distribution” model, that has been trained on WAD injury labels directly. We conduct this comparison on different datasets and models.

In App. C.2, we evaluate the NT5-Clf model in a classification formulation and report accuracy. As expected, we find that accuracy decreases in every setting with performance drops between 0.001% and 0.3%. In Fig. 15, we evaluate the NT5-Reg model in a regression setting measured in MSE. We find that the performance decreases in the out-of-distribution settings as evidenced by an average increase of 1.12 in MSE. Finally, in Fig. 16, we turn to an NT5-Gen model in a generative setting. As an evaluation metric, we consider *Exact-Match* and observe a decrease of 0.18 in *Exact-Match* on average.

8 Application: Overlooked Victim Types

Most event datasets feature only one column detailing victim counts. This column typically quantifies fatalities, as they are considered least ambiguous and most important (Kalyvas, 2006; Chaudoin et al., 2017). The Armed Conflict Location & Event Data Project (ACLED) (Raleigh et al., 2010; Raleigh and Kishi, 2019) recently published curated datasets containing violence against health-care workers, media personnel, and women. Con-

sidering the ACLED dataset on Political Violence Targeting Women & Demonstrations Featuring Women, we find that more than 85% of events have *zero* fatalities. This means, many forms of violence remain non-quantified, often those against “marginalized” groups of society.

Using the methods presented in this work, we can extract much more fine-grained victim types such as “injured women” and “abducted women”. To this end, we rely on the NT5-Gen model that we fine-tuned on the NAVCO data, without specifically asking for “women”. In Fig. 4, we present exemplary two-month time series of events in Syria. We find that our model has a higher recall than precision on the ground truth annotations for fatality counts. This may be desirable since we would like to avoid overlooking true victim counts.

9 Discussion

This work surveys different task formulations of victim count extraction and inspects desiderata like accuracy, reliability, and robustness of different models. We now summarize our findings and conclude which approach performs best under which circumstances (Tab. 5).

Some of the parsing-based approaches have the advantage of requiring no ground truth annotations of the extracted victim counts. This means, there is no need for training, but instead, a manually curated list of patterns and rules has to be assembled. The regex approach, for instance, has minimum requirements regarding hardware, but writing regex patterns is very time-intensive and can be prone to mistakes. Overall, the baseline models shine when it comes to speed, and they perform reasonable when victim counts are explicitly mentioned. Yet they fail at complex reasoning. For instance, when asking for the count of deaths in “one child and four women lost their lives”, all baselines mistakenly output “1”.

This is where language model-based methods have a competitive edge. The fine-tuned NT5-Gen model has high accuracy both in *Exact-Match* metric and relative error metric. Surprisingly, it is also well-calibrated and relatively robust in the few-shot and out-of-distribution setting. This performance comes at the costs of reduced speed, the requirement of large amounts of training data, and the need for resources like GPUs to be deployed on a large scale.

Comparing classification and regression objec-

tives, we conclude that classification is easier to handle. In most settings, it may be sufficient to extract a range rather than an exact number anyways. In comparison to generation, in classification and regression settings, models show higher calibration errors and require post-hoc calibration to adjust the model confidence.

10 Related Works

This work interfaces with related works from different disciplines to improve the measurement of crisis intensity. It draws inspiration from recent advancements in question answering models with a focus on numbers and math word problems. This includes number-enhanced language models more generally. Our work also connects with model calibration in natural language processing (NLP) more generally.

Measurement of Crisis Intensity. Extracting information about crises has been widely explored using social media data (Temnikova et al., 2015) and newspapers (Keith et al., 2017; Halterman et al., 2021). Most existing measures of crisis intensity focus on counts of event types (Goldstein, 1992; Terechshenko, 2020; Stoehr et al., 2022) or fatality counts (Kalyvas, 2006). Previous work studies friend-enemy relationships (Han et al., 2019; Russo et al., 2022a; Stoehr et al., 2021, 2023) and conflict-indicative changes in word embeddings (Kutuzov et al., 2017).

Numerical Question Answering. Numerical Question Answering pertains to the task of providing numeric answers to questions. An exemplary model is NAQANet (Dua et al., 2019), which extends QANet (Yu et al., 2018) with numerical operations. Neural Module Networks (Gupta et al., 2020) learn and execute a chain of logical learnable and differentiable modules. Some of these modules are specifically targeted at mathematical operations such `find-num` or `count`. Other approaches leverage knowledge graphs (Davidov and Rappoport, 2010; Kotnis and García-Durán, 2019) or graph neural networks (Chen et al., 2020). Thawani et al. (2021) provides a detailed overview over methods for representing and modeling numbers in NLP.

Number-enhanced Language Models. More recent work in number question answering relies on pre-trained large language models. GenBERT (Geva et al., 2020) improves numeric reasoning abilities by including a large amount of

	Accuracy Optimization			Reliability		Robustness		Hardware
	Absolute Error	Relative Error	String Match		Need Training	Stable in OOD		
REGEX	High	Medium	Medium	N/A	No	N/A	Low	
DEP	Medium	High	Low-Medium	N/A	No	N/A	Low	
SRL	Low-Medium	Medium	High	N/A	No	N/A	Low-Medium	
CLF	N/A	N/A	N/A	Low	Medium-High	Low	Medium - High	
REG	Low	Low	N/A	Low	High	Low-Medium	Medium - High	
GEN	Low	Low	Medium-High	High	Low-Medium	Medium-High	High	

Table 5: Overview of pros and cons of different models. We list baselines: regular expressions (REGEX), dependency parsing (DEP), and semantic role labeling (SRL). The CLF, REG, GEN refer to the fine-tuned NT5-Clf, NT5-Reg, and NT5-Gen models. Absolute / Relative Error pertains to the absolute/relative error between true victim counts and model predictions taking the real numerical value of the counts (e.g., mean squared error). String Match considers string metrics like *Exact-Match* used in question answering. The reliability column is based on experiments in model calibration. Robustness is divided into the need for training on a large annotated dataset and the stability in out-of-distribution (OOD). N/A means “Not Applicable”.

synthetic data containing numbers. Codex (Chen et al., 2021) and NT5 (Yang et al., 2021) apply similar strategies and are trained on code and math word problems. Other approaches focus on step-by-step reasoning such as Minerva (Lewkowycz et al., 2022), scratchpad (Nye et al., 2021) and chain-of-thought prompting (Wei et al., 2022). Lefebvre and Stoehr (2022) propose a prompting-based method particularly for conflict event classification.

Calibration of NLP Models. The calibration of NLP models has been extensively studied in classification (Guo et al., 2017) and structured prediction tasks (Kuleshov and Liang, 2015; Nguyen and O’Connor, 2015). Calibration methods have been adapted in language modeling (Braverman et al., 2020; Kong et al., 2020), question answering (Kamath et al., 2020; Jiang et al., 2021), and machine translation (Kumar and Sarawagi, 2019; Wang et al., 2020).

11 Conclusion

We presented *victim count extraction*, a challenging and impactful task. The task can be tackled using different formulations and models. Models should be evaluated along different dimensions such as accuracy, reliability, and robustness. We survey this ambiguity of victim count extraction, identify promising directions, and discuss outlooks and applications.

Limitations

The models may be biased or reproduce biases inherent in their training data. Presenting unrelated, faulty or immoral questions to a model can cause

unguided and malicious behavior. For example, we caution of asking questions such as “How many people *will be injured...?*”; and even worse “How many people *should be injured...?*”. Improving model calibration will help defending against these issues and enable awareness of when to abstain from answering.

Ethics Statement

This work originated from the motivation to diversify victim count extraction towards underrepresented victim types and overlooked forms of violence. This work ultimately intends to assist researchers and analysts in the sector of humanitarian aid who are in demand of accurate victim count information.

References

- Mark Braverman, Xinyi Chen, Sham Kakade, Karthik Narasimhan, Cyril Zhang, and Yi Zhang. 2020. *Calibration, entropy rates, and memory in language models*. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1089–1099. PMLR.
- Stephen Camilleri, Joel Azzopardi, and Matthew R. Agius. 2019. *Investigating the relationship between earthquakes and online news*. In *2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, page 203–210.
- Stephen Chaudoin, Zachary Peskowitz, and Christopher Stanton. 2017. *Beyond zeroes and ones: The intensity and dynamics of civil conflict*. *The Journal of Conflict Resolution*, 61(1):56–83. Publisher: Sage Publications, Inc.

- Kunlong Chen, Weidi Xu, Xingyi Cheng, Zou Xiaochuan, Yuyu Zhang, Le Song, Taifeng Wang, Yuan Qi, and Wei Chu. 2020. [Question directed graph attention network for numerical reasoning over text](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6759–6768, Online. Association for Computational Linguistics.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebbgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. [Evaluating large language models trained on code](#). In *arXiv*, volume 2107.03374.
- Maarten Cruyff, Jan van Dijk, and Peter G. M. van der Heijden. 2017. [The challenge of counting victims of human trafficking: Not on the record: A multiple systems estimation of the numbers of human trafficking victims in the Netherlands in 2010–2015 by year, age, gender, and type of exploitation](#). *CHANCE*, 30(3):41–49.
- Patricia Cullen, Myrna Dawson, Jenna Price, and James Rowlands. 2021. [Intersectionality and invisible victims: Reflections on data challenges and vicarious trauma in femicide, family and intimate partner homicide research](#). *Journal of Family Violence*, 36(5):619–628.
- James Darcy and Charles-Antoine Hofmann. 2003. [According to need? Needs assessment and decision-making in the humanitarian sector](#). Technical report, Overseas Development Institute.
- Dmitry Davidov and Ari Rappoport. 2010. [Extraction and approximation of numerical attributes from the web](#). In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, page 1308–1317, Uppsala, Sweden. Association for Computational Linguistics.
- Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. [Autoregressive entity retrieval](#). In *9th International Conference on Learning Representations (ICLR)*, Virtual Event, Austria.
- Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. [DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2368–2378, Minneapolis, Minnesota. Association for Computational Linguistics.
- Lars Döhling and Ulf Leser. 2011. [EquatorNLP: Pattern-based information extraction for disaster response](#). In *Terra Cognita 2011 Workshop, Foundations, Technologies and Applications of the Geospatial Web*, volume 38, Bonn, Germany.
- Mor Geva, Ankit Gupta, and Jonathan Berant. 2020. [Injecting numerical reasoning skills into language models](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 946–958, Online. Association for Computational Linguistics.
- Joshua Goldstein. 1992. [A conflict-cooperation scale for WEIS events data](#). *The Journal of Conflict Resolution*, 36(2):369–385.
- Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. [On calibration of modern neural networks](#). In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML’17, page 1321–1330, Sydney, Australia. JMLR.org.
- Nitish Gupta, Kevin Lin Lin, Dan Roth, Sameer Singh, and Matt Gardner. 2020. [Neural module networks for reasoning over text](#). In *8th International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia.
- Andrew Halterman, Katherine Keith, Sheikh Sarwar, and Brendan O’Connor. 2021. [Corpus-level evaluation for event QA: The IndiaPoliceEvents corpus covering the 2002 Gujarat violence](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP*, pages 4240–4253, Online.
- Xiaochuang Han, Eunsol Choi, and Chenhao Tan. 2019. [No permanent friends or enemies: Tracking relationships between nations from news](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 1660–1676.
- Sebastian Hellmeier, Nils B. Weidmann, and Espen Geelmuyden Rød. 2018. [In the spotlight: Analyzing sequential attention effects in protest reporting](#). *Political Communication*, 35(4):587–611. Publisher: Routledge.
- Muhammad Imran, Shady Elbassuoni, Carlos Castillo, Fernando Diaz, and Patrick Meier. 2013. [Practical extraction of disaster-relevant information from social media](#). In *Proceedings of the 22nd International Conference on World Wide Web*, page 1021–1024, Rio de Janeiro, Brazil. ACM.

- Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How can we know when language models know? On the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.
- JRC Science Hub. 2018. European media monitor (EMM). Technical report, Joint Research Center Science Hub.
- Stathis N. Kalyvas. 2006. *The Logic of Violence in Civil War*. Cambridge University Press, Cambridge.
- Amita Kamath, Robin Jia, and Percy Liang. 2020. Selective question answering under domain shift. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5684–5696, Online. Association for Computational Linguistics.
- Katherine Keith, Abram Handler, Michael Pinkham, Cara Magliozzi, Joshua McDuffie, and Brendan O’Connor. 2017. Identifying civilians killed by police with distantly supervised entity-event extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark. Association for Computational Linguistics.
- Lingkai Kong, Haoming Jiang, Yuchen Zhuang, Jie Lyu, Tuo Zhao, and Chao Zhang. 2020. Calibrated language model fine-tuning for in- and out-of-distribution data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1326–1340, Online. Association for Computational Linguistics.
- Bhushan Kotnis and Alberto García-Durán. 2019. Learning numerical attributes in knowledge bases. In *1st Conference on Automated Knowledge Base Construction, AKBC 2019, Amherst, MA, USA, May 20-22, 2019*.
- Keith Krause. 2013. Challenges to counting and classifying victims of violence in conflict, post-conflict, and non-conflict settings. In *Counting Civilian Casualties: An Introduction to Recording and Estimating Nonmilitary Deaths in Conflict*. Oxford University Press.
- Tino Kreutzer, Patrick Vinck, Phuong N. Pham, Aijun An, Lora Appel, Eric DeLuca, Grace Tang, Muath Alzghool, Kusum Hachhethu, Bobi Morris, Sandie L. Walton-Ellery, John Crowley, and James Orbinski. 2020. Improving humanitarian needs assessments through natural language processing. *IBM Journal of Research and Development*, 64(1/2):9:1–9:14.
- Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. 2018. Accurate uncertainties for deep learning using calibrated regression. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2796–2804. PMLR.
- Volodymyr Kuleshov and Percy S Liang. 2015. Calibrated structured prediction. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.
- Aviral Kumar and Sunita Sarawagi. 2019. Calibration of encoder decoder models for neural machine translation. In *arXiv*, volume 1903.00802.
- Andrey Kutuzov, Erik Velldal, and Lilja Øvrelid. 2017. Tracing armed conflicts with diachronic word embedding models. In *Proceedings of the Events and Stories in the News Workshop*, pages 31–36.
- Clement Lefebvre and Niklas Stoehr. 2022. Rethinking the event coding pipeline with prompt entailment. In *arXiv*, volume 2210.05257.
- Christian Leibig, Vaneeda Allken, Murat Seçkin Ayhan, Philipp Berens, and Siegfried Wahl. 2017. Leveraging uncertainty information from deep neural networks for disease detection. *Scientific Reports*, 7(1):17816.
- Raphael Lepuschitz and Niklas Stoehr. 2021. SeismographAPI: Visualising temporal-spatial crisis data. In *arXiv*, volume 2107.12443.
- Orion A. Lewis, Erica Chenoweth, and Jonathan Pinckney. 2016. Nonviolent and violent campaigns and outcomes 3.0: Effects of tactical choices on strategic outcomes codebook. Technical Report Version 0.9b5, Sie Cheou-Kang Center for International Security and Diplomacy University of Denver.
- Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. 2022. Solving quantitative reasoning problems with language models. In *arXiv*, volume 2206.14858.
- Khanh Nguyen and Brendan O’Connor. 2015. Posterior calibration and exploratory analysis for natural language processing models. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1587–1598, Lisbon, Portugal. Association for Computational Linguistics.
- Clayton Norris, Philip Schrodt, and John Beieler. 2017. PETRARCH2: Another event coding program. *The Journal of Open Source Software*, 2(9).
- Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. 2021. Show your work: Scratchpads for intermediate computation with language models. In *arXiv*, volume 2112.00114.
- Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

- Benjamin J. Radford. 2021. [Automated dictionary generation for political eventcoding](#). *Political Science Research and Methods*, 9(1):157–171.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text Transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.
- Clionadh Raleigh and Roudabeh Kishi. 2019. [Comparing conflict data – Similarities and differences across datasets](#). ACLED Report, The Armed Conflict Location & Event Data Project (ACLED).
- Clionadh Raleigh, Andrew Linke, Håvard Hegre, and Joakim Karlsen. 2010. [Introducing ACLED-armed conflict location and event data](#). *Journal of Peace Research*, 47(5):651–660.
- Subhro Roy, Tim Vieira, and Dan Roth. 2015. [Reasoning about quantities in natural language](#). *Transactions of the Association for Computational Linguistics*, 3:1–13.
- Koustav Rudra, Niloy Ganguly, Pawan Goyal, and Saptarshi Ghosh. 2018. [Extracting and summarizing situational information from the Twitter social media during disasters](#). *ACM Transactions on the Web*, 12(3):1–35.
- Giuseppe Russo, Christoph Gote, Laurence Brandenberger, Sophia Schlosser, and Frank Schweitzer. 2022a. [Disentangling active and passive cosponsorship in the U.S. congress](#). In *arXiv*, volume 2205.09674.
- Giuseppe Russo, Manoel Horta Ribeiro, Giona Casiraghi, and Luca Verginer. 2022b. [Understanding online migration decisions following the banning of radical communities](#). In *arXiv*, volume 2212.04765.
- Philip A. Schrodt. 2009. [TABARI: Textual analysis by augmented replacement instructions](#). Technical Report 0.7, Parus Analytics.
- Philip A. Schrodt and Jay Ulfelder. 2016. [Political instability task force atrocities event data collection codebook](#). Technical Report Version 1.1b1, Parus Analytics.
- Hao Song, Tom Diethe, Meelis Kull, and Peter Flach. 2019. [Distribution calibration for regression](#). In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5897–5906. PMLR.
- Ralf Steinberger, Martin Atkinson, Teofilo Domingo, Erik van der Goot, Jens Linge, Charles Macmillan, Hristo Tanev, and Gerhard Wagner. 2017. [EMM: Supporting the Analyst by Turning Multilingual Text into Structured Data](#). Erich Schmidt Verlag, Berlin, Germany.
- Niklas Stoehr, Lucas Torroba Hennigen, Josef Valvoda, Robert West, Ryan Cotterell, and Aaron Schein. 2022. [An ordinal latent variable model of conflict intensity](#). In *arXiv*, volume 2210.03971.
- Niklas Stoehr, Benjamin J. Radford, Ryan Cotterell, and Aaron Schein. 2023. [The Ordered Matrix Dirichlet for modeling ordinal dynamics](#). In *arXiv*, volume 2212.04130.
- Niklas Stoehr, Lucas Torroba Hennigen, Samin Ahbab, Robert West, and Ryan Cotterell. 2021. [Classifying dyads for militarized conflict analysis](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7775–7784, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Irina P. Temnikova, Carlos Castillo, and Sarah Vieweg. 2015. [EMTerms 1.0: A terminological resource for crisis tweets](#). In *12th Proceedings of the International Conference on Information Systems for Crisis Response and Management*, Krystiansand, Norway. ISCRAM Association.
- Zhanna Terechshenko. 2020. [Hot under the collar: A latent measure of interstate hostility](#). *Journal of Peace Research*, 57(6):764–776.
- Avijit Thawani, Jay Pujara, Filip Ilievski, and Pedro Szekely. 2021. [Representing numbers in NLP: a survey and a vision](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–656, Online. Association for Computational Linguistics.
- Jack Vincent. 1979. [Project theory](#). *Lanham M.D. University Press of America*.
- Shuo Wang, Zhaopeng Tu, Shuming Shi, and Yang Liu. 2020. [On the inference calibration of neural machine translation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3070–3079, Online. Association for Computational Linguistics.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. [Chain-of-thought prompting elicits reasoning in large language models](#). In *arXiv*, volume 2201.11903.
- David Widmann, Fredrik Lindsten, and Dave Zachariah. 2021. [Calibration tests beyond classification](#). In *9th International Conference on Learning Representations (ICLR)*, Virtual Event, Austria.
- Peng-Jian Yang, Ying Ting Chen, Yuechan Chen, and Daniel Cer. 2021. [NT5?! Training T5 to perform numerical reasoning](#). In *arXiv*, volume 2104.07307.
- Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao Zhao, Kai Chen, Mohammad Norouzi Norouzi, and Quoc V. Le. 2018. [QANet: Combining](#)

local convolution with global self-attention for reading comprehension. In *6th International Conference on Learning Representations (ICLR)*, Vancouver, BC, Canada.

Vanni Zavarella, Jakub Piskorski, Camelia Ignat, Hristo Tanev, and Martin Atkinson. 2020. [Mastering the media hype: Methods for deduplication of conflict events from news reports](#). In *Proceedings of the firstings of the first AI4Narratives Workshop*.

Yunfeng Zhang, Q. Vera Liao, and Rachel K. E. Bellamy. 2020. [Effect of confidence and explanation on accuracy and trust calibration in AI-assisted decision making](#). In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, FAT* ’20*, page 295–305, New York, NY, USA. Association for Computing Machinery.

A Regex Patterns

We convert any non-digitized numeral expressions into a digitized format (e.g. twelve → 12). Regex patterns are designed for both passive and active voices. We also distinguish plural (“are” and “were”) and singular forms (“is”, “was”) for passive voice patterns. The algorithm checks with the following order: passive plural, passive single, and active. If multiple numbers are extracted, the first is kept. We list the regex patterns used to extract victim counts in Tab. 6, for death counts and injury counts respectively.

B Accuracy Evaluation

In this section, we complement the accuracy evaluation of the models in §5.

B.1 Exact-Match and F_1 score on Death Counts

The *Exact-Match* and F_1 scores on extracting the death counts are shown in Tab. 7, which compares the performance of the baseline models and the fine-tuned NT5-Gen model. Similar to the results on the injury counts Tab. 1, the fine-tuned NT5-Gen model performs better than all baselines and the SRL has the best accuracy among baselines.

B.2 Confusion Matrix on Death Counts

Similar to Fig. 1 shown in §5.1, Fig. 5 plots the confusion matrices of the binned death counts for the different datasets, which compare the accuracy of the baseline models with the fine-tuned NT5-Gen model.

B.3 Results on Classification and Regression

In §5.2, we have shown the results of the NT5-Clf model and the NT5-Reg model fine-tuned on NAVCO injury counts in Tab. 3.

Here, we use the same metrics and display the classification performance on other datasets. In specific, Tab. 8, Tab. 9, Tab. 10, Tab. 11, and Tab. 12 respectively show the classification performance of the NT5-Clf model fine-tuned on WAD death counts, WAD injury counts, NAVCO death counts, EMM death counts, and EMM injury counts.

Similarly, we provide the scatter plots of the fine-tuned NT5-Reg models initialized with different pre-trained weights in this section: WAD death counts (Fig. 6), WAD injury counts (Fig. 7), NAVCO death counts (Fig. 8), EMM death counts (Fig. 9), and EMM injury counts (Fig. 10).

C Robustness Evaluation

In this section, we provide the detailed performance of the few-shot setting (App. C.1) and the out-of-distribution setting (App. C.2) discussed in §7.

C.1 Few-shot Performance

We display the results of the few-shot settings where different proportions of the training set are used to fine-tune the models. For each formulation, the left figure is the variation of the accuracy metrics and the right figure is the variation of the calibration error. Fig. 11, Fig. 12, and Fig. 13 are performance of the few-shot settings of the fine-tuned NT5-Clf, NT5-Reg, and NT5-Gen models respectively.

With respect to accuracy metrics, the classification accuracy and the F_1 score is plotted for the fine-tuned NT5-Clf model.

For the regression, we plot the change of mean squared error on the log transformed counts. In addition, we plot the pinball losses using two targeting quantile (at 10% and at 90%).

Lastly, the *Exact-Match* and the F_1 scores are drawn for the fine-tuned NT5-Gen model.

C.2 Out-of-distribution Setting

For each task formulation, we examine the accuracy performance in the out-of-distribution setting for the fine-tuned NT5-Clf (Fig. 14), NT5-Reg (Fig. 15), and NT5-Gen (Fig. 16). For all plots, the x-axis is the accuracy metric used in each task formulation, and the y-axis indicates the test set to be made inferences on. The red bar indicates the

Data Type	Regex Type	Regex Pattern
Death	Passive Plural	\d(\d ,)*(?!D*(injur wound))(?=.*(\b(were are)\D*\b(killed dead died slain)))
	Passive Singular	\S*(?!D*(injur wound))(?=.*(\b(was is)\D*\b(killed dead died slain)))
	Active	(kill slay slain)\D*\b\d(\d ,)*
Injury	Passive Plural	\d(\d ,)*(?!.*(\b(were are)?\D*\b(killed dead died slain)))(?=.*\b(injur wound))
	Passive Singular	\S*(?=(was is).*\b(injur wound))(?!D*(\b(were are)\D*\b(killed dead died slain)))
	Active	(injured? wound)\D*\d+

Table 6: Regex patterns.

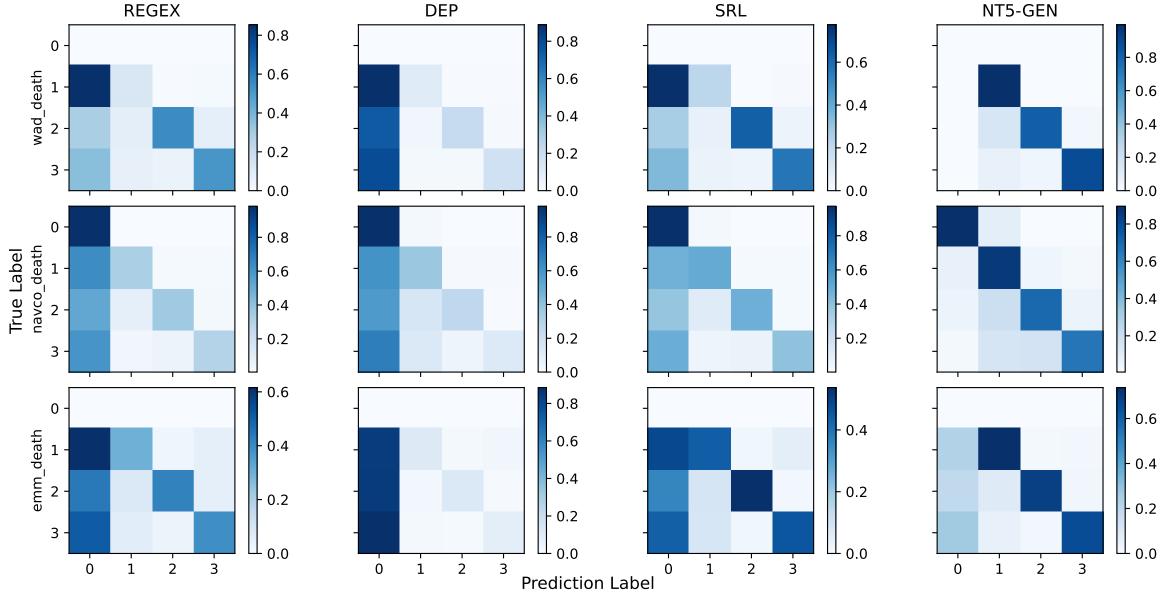


Figure 5: Confusion matrices of baseline models and fine-tuned NT5-Gen model (columns) of extracting death counts from different data (rows). We convert the true victim counts and model predictions into 4 categories: for any count y , “0” is $y = 0$, “1” is $0 < y \leq 3$, “2” is $3 < y \leq 10$ and “3” is $y > 10$. Values are normalized over true counts.

	Exact Match			F_1		
	WAD	NAVCO	EMM	WAD	NAVCO	EMM
Regex	0.3543	0.3921	0.2835	0.3897	0.4196	0.3242
Dep	0.1506	0.3526	0.0767	0.2064	0.3792	0.1317
SRL	0.4342	0.4839	0.3972	0.7794	0.4837	0.3613
NT5	0.6798	0.6590	0.6322	0.8458	0.5436	0.4917

Table 7: *Exact-Match* and F_1 scores of the baseline models and the fine-tuned NT5-Gen model on death counts. Best metrics are **bolded**. DEP refers to the dependency parsing model and SRL refers to the semantic role labeling model.

performance of in-distribution performance, e.g., accuracy of WAD death test data using the model fine-tuned on WAD death.

With respect to the accuracy metric, different formulations use their corresponding metric. For the classification setting, we show the variation in classification accuracy. For the regression setting, we show the variation in mean squared errors.

	Accuracy	F1 score	Precision	Recall
NT5	0.81	0.81	0.80	0.83
T5	0.81	0.81	0.81	0.84
BERT	0.86	0.86	0.86	0.88

Table 8: Classification results on WAD death counts with the NT5-Clf model initialized by different pre-trained weights: NT5, T5-SMALL, and BERT-BASE-UNCASED. F_1 , precision and recall scores are macro.

	Accuracy	F1 score	Precision	Recall
NT5	0.77	0.69	0.70	0.69
T5	0.76	0.69	0.70	0.68
BERT	0.93	0.91	0.91	0.90

Table 9: Classification results on WAD injury counts with the NT5-Clf model initialized by different pre-trained weights: NT5, T5-SMALL, and BERT-BASE-UNCASED. F_1 , precision and recall scores are macro.

	Accuracy	F1 score	Precision	Recall
NT5	0.65	0.60	0.62	0.59
T5	0.65	0.60	0.61	0.59
BERT	0.52	0.23	0.17	0.33

Table 10: Classification results on NAVCO death counts with the NT5-Clf model initialized by different pre-trained weights: NT5, T5-SMALL, and BERT-BASE-UNCASED. F_1 , precision and recall scores are macro.

	Accuracy	F1 score	Precision	Recall
NT5	0.72	0.65	0.66	0.65
T5	0.70	0.63	0.65	0.63
BERT	0.84	0.80	0.82	0.78

Table 11: Classification results on EMM death counts with the NT5-Clf model initialized by different pre-trained weights: NT5, T5-SMALL, and BERT-BASE-UNCASED. F_1 , precision and recall scores are macro.

	Accuracy	F1 score	Precision	Recall
NT5	0.68	0.58	0.60	0.57
T5	0.68	0.58	0.59	0.57
BERT	0.81	0.77	0.79	0.76

Table 12: Classification results on EMM injury counts with the NT5-Clf model initialized by different pre-trained weights: NT5, T5-SMALL, and BERT-BASE-UNCASED. F_1 , precision and recall scores are macro.

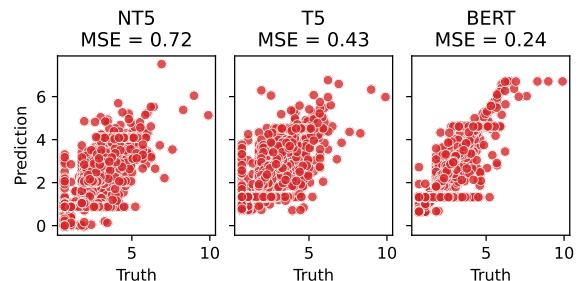


Figure 6: Scatter plots of the fine-tuned NT5-Reg model initialized with different pre-trained weights (NT5, T5-SMALL, and BERT-BASE-UNCASED). The models are trained on log-transformed victim counts of WAD death.

For the generation setting, we show the change in *Exact-Match* scores.

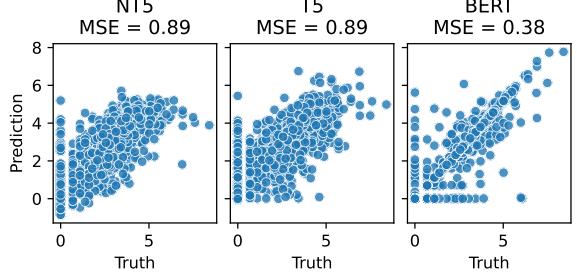


Figure 7: Scatter plots of the fine-tuned NT5-Reg model initialized with different pre-trained weights (NT5, T5-SMALL, and BERT-BASE-UNCASED). The models are trained on log-transformed victim counts of WAD injury.

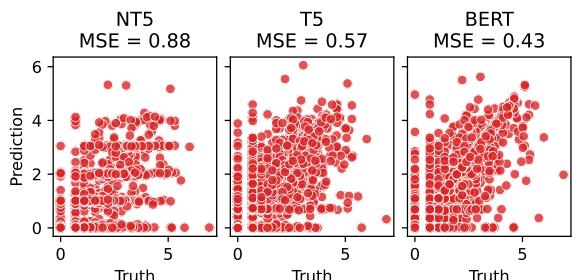


Figure 8: Scatter plots of the fine-tuned NT5-Reg model initialized with different pre-trained weights (NT5, T5-SMALL, and BERT-BASE-UNCASED). The models are trained on log-transformed victim counts of NAVCO death.

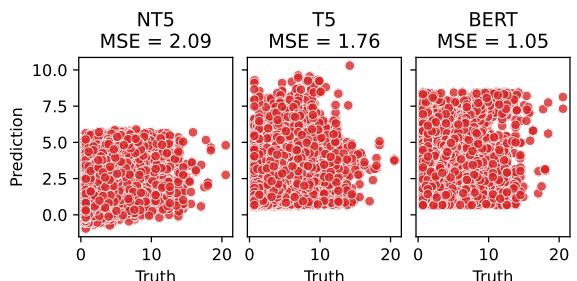


Figure 9: Scatter plots of the fine-tuned NT5-Reg model initialized with different pre-trained weights (NT5, T5-SMALL, and BERT-BASE-UNCASED). The models are trained on log-transformed victim counts.

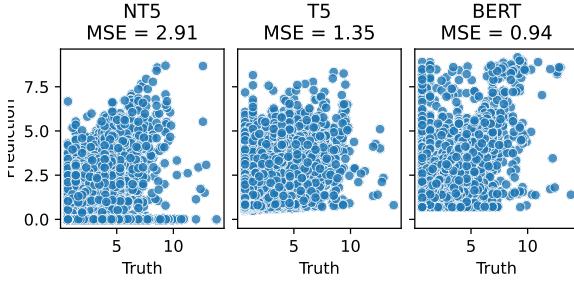


Figure 10: Scatter plots of the fine-tuned NT5-Reg model initialized with different pre-trained weights (NT5, T5-SMALL, and BERT-BASE-UNCASED). The models are trained on log-transformed victim counts.

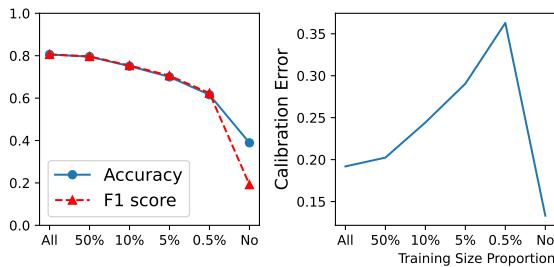


Figure 11: Few-shot performance of the fine-tuned classification model on WAD death counts.

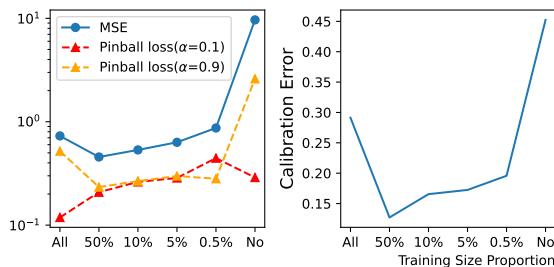


Figure 12: Few-shot performance of the fine-tuned regression model on WAD death counts.

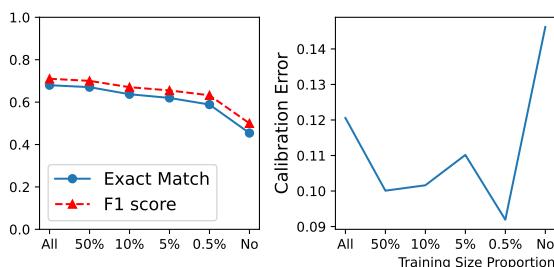


Figure 13: Few-shot performance of the fine-tuned NT5-Gen model on WAD death counts.

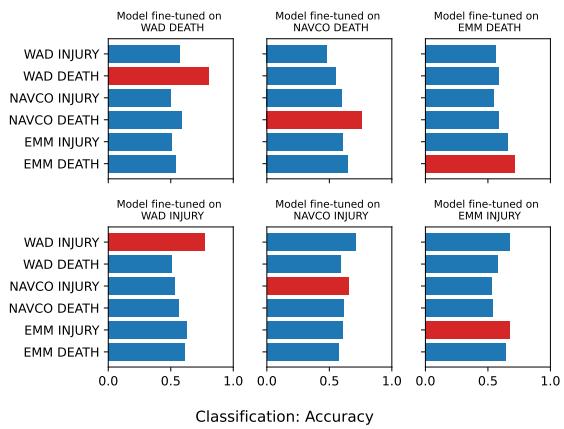


Figure 14: Classification accuracy for using the fine-tuned NT5-Clf models on out-of-distribution data (blue) and on in-distribution data (red)

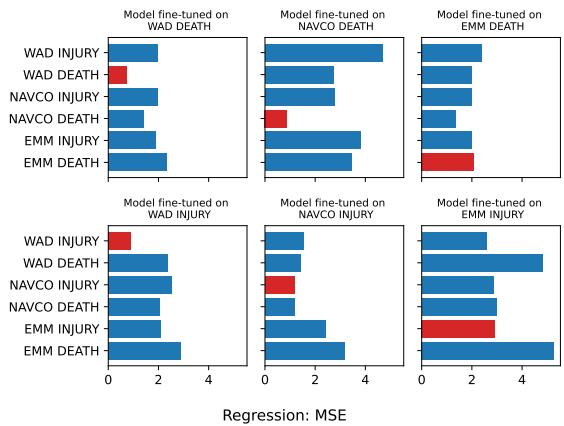


Figure 15: Mean squared error for using the fine-tuned NT5-Reg models on out-of-distribution data (blue) and on in-distribution data (red)

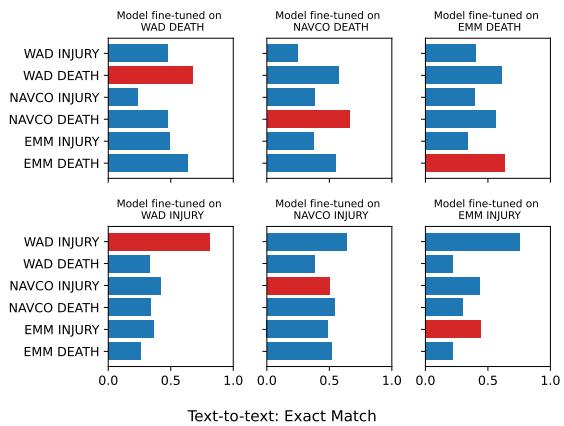


Figure 16: Exact-Match for using the fine-tuned NT5-Gen models on out-of-distribution data (blue) and on in-distribution data (red)