AllegatorTrack: Combining and Reporting Results of Truth Discovery from Multi-source Data

Dalia Attia Waguih, Naman Goel, Hossam M. Hammady, Laure Berti-Equille

Qatar Computing Research Institute
Doha, Qatar
{dattia,ngoel,hhammady,lberti}@qf.org.qa

I. Motivation

In the Web, a massive amount of user-generated contents is available through various channels, e.g., texts, tweets, Web tables, databases, multimedia-sharing platforms, etc. Conflicting information, rumors, erroneous and fake contents can be easily spread across multiple sources, making it hard to distinguish between what is true and what is not. How do you figure out that a lie has been told often enough that it is now considered to be true? How many lying sources are required to introduce confusion in what you knew before to be the truth? To answer these questions, we present AllegatorTrack, a system that discovers true claims among conflicting data from multiple sources.

Our work falls under the recently emerging research field of computational journalism, where recent work, e.g., [11], [5], [1] tackles the problem of fact-checking and ascertaining the veracity of online information. As shown by our recent extensive comparative study [9], current methods generally suffer from several drawbacks: opacity, complex parameter setting, scalability and repeatability issues, and provide results that are difficult to interpret.

The goal of AllegatorTrack is to provide users with a system and API to test existing truth discovery computation methods, combine their results, provide explanations of the truth discovery results and allow the users to generate allegations.

In this demo, we will present AllegatorTrack whose architecture is illustrated in Figure 1 and focus on its truth discovery computation and reporting modules (in red). We will showcase AllegatorTrack key features for reporting truth discovery results, explanations, and allegations.

II. AllegatorTrack Overview

Given a set of assertions claimed by multiple sources, the ultimate goal of online truth discovery is to label each claimed information as true or false and compute the reliability and truthfulness of its respective source. Various probabilistic models have been proposed to iteratively compute and update the trustworthiness of a source as a function of the belief in its claims, and then the belief score of each claim as a function of the trustworthiness of the sources asserting it (e.g., TruthFinder [12]). Some truth discovery models have incorporated prior knowledge either about the source reputation or self-confidence in its assertions (e.g., LCA models [8]). Beyond source trustworthiness and claim belief, other aspects have been considered for truth discovery computation: the dependence between sources (e.g., Depen models [1]), the temporal dimension in discovering evolving truth [3], the difficulty of ascertaining the veracity of certain claims (e.g., Cosine, 2- and 3-Estimates [4]), and the management of negative claims (e.g., LTM [13]) or Boolean claims (e.g., MLE [10]). However, in truth discovery scenarios, it is common that the user wants to understand not only the labeling results (*i.e.*, classification of the claims as true or false) but also how the trustworthiness scores of the sources have been computed, and finally, how the results corroborate (or not) the *a priori* opinion he/she may have on the credibility or authoritativeness of the sources.

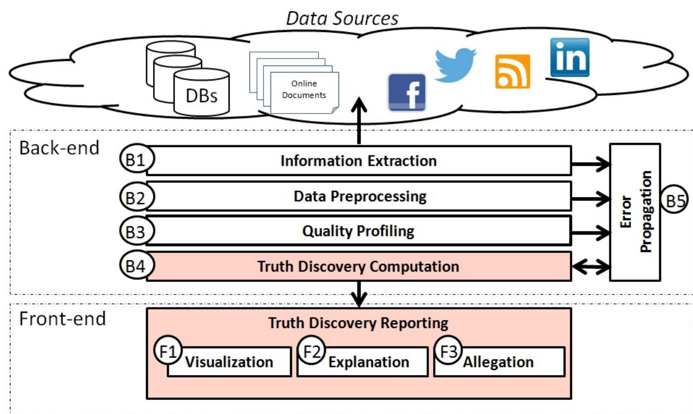

Fig. 1: Architecture of the AllegatorTrack system

There is also a need for "what-if" or "why-not" analysis, a feature that is commonly sought for in many data analysis applications and which is as important as the need for reverse-engineering vague claims and finding counterarguments [11]. As a matter of fact, none of the previous approaches have explored how to explain truth discovery results in a comprehensive manner. AllegatorTrack extends previous work with the ability to report results from twelve fact-checking models and allows the user to generate explanations and allegations. Allegation can be considered as another kind of explanation by intervention since there exists a minimal number of updated or new claims that can change any truth discovery computation results, making false claims become true (and vice versa).

III. Architecture

Truth discovery from user-generated contents is a complex iterative process including various tasks: selection of heterogeneous data sources, information and context extraction from structured, semi- and non-structured contents, data integration (including formatting, cleaning, entity resolution, and fusion), and evidence-based fact verification. In Figure 1, the back-end of AllegatorTrack extracts data from various data sources (B1); it performs data preprocessing (B2), determines data and source quality indicators (B3) and computes the confidence of the data values claimed by each source (B4) and revises the source truthfulness scores iteratively. At each stage of the truth discovery process, errors can be introduced, e.g., information extractors may provide uncertain results as well as entity resolution and uncertainty has propagated in the truth computation (B5). The front-end provides an interface to the user for searching the truth discovery results, generating Sankay diagrams for visualization (F1), generating explanations (F2), and allegations (F3) on users' demand. We implemented an optimized version of the AllegatorTrack system in Java version 7 and Ruby on Rails. The graphical user interface was created to allow users to specify parameters for multiple truth discovery scenarios, select and run multiple truth discovery models, explore and combine their results, get explanations and generate allegations as we will show in the demonstration. The demonstration will not cover the information extraction and preprocessing stages.

IV. Key Features

The claims are the assertions made by multiple sources (and whose veracity is unknown) and they are organized into data items that are disjoint mutual exclusion sets as defined in [7] referring to a feature of one real-world entity, e.g., the place of birth of a person in the Biography data set, the number of deaths of World War 2 or the list of the author names of a particular book as presented in Table 1. One or more claims (uniquely identified by a claim identifier) are associated with one data item identifier. Only one value is assumed to be true. Claims can be either positive or negative. Cases such as source "S claims that A is false" or "S does not claim A is true" can be considered. But indirect source attribution are not supported, e.g., "S_1 claims that S_2 claims that A is true".

A source is not supposed to contribute uniformly to all the claims it expresses and one goal of AllegatorTrack is to profile the trustworthiness of each source since it can be computed by all the algorithms and normalized. Sources are not necessarily independent and AllegatorTrack can compute the source dependency as defined by [1]. Finally, each claim is assumed to be either true or false. AllegatorTrack computes and manages the trustworthiness scores of the sources and the confidence scores of each claim and the truth discovery labels.

The key features of AllegatorTrack that will be demonstrated are the following:

Multiple Truth Discovery Models. AllegatorTrack supports twelve truth discovery models from the literature, namely: TruthFinder [12], Cosine, 2-Estimates and 3-Estimates [4], Depen with its four variants [2], SimpleLCA and GuessLCA [8], MLE [10], and LTM [13]. AllegatorTrack enables the user to explore the results of existing truth discovery models to understand their differences and limitations. When the ground truth is available, it also provides quality measures of the models in terms of precision, recall, accuracy, and specificity. The models can also be executed through API at `dafna.qcri.org`. Specific transformations of the data set are handled for executing LTM and MLE models. In these cases, multi-valued claims (e.g., list of authors) are automatically split into multiple mono-valued claims.

Collective Inference. We have observed that none of the truth discovery methods constantly outperforms the others in terms of precision, accuracy, recall, and specificity [9]. A "one-fits-all" approach is hardly achievable to handle various data set characteristics and truth discovery scenarios. Moreover, a complete ground truth data set is rarely available to measure objectively the quality performance of the truth discovery methods. To address these issues, AllegatorTrack combines the results of multiple methods with Bayes combination. Moreover, it applies collective inference for computing a final truth discovery result: it exploits the relational autocorrelation between the truth labels of various models and takes advantage of the relational data characteristic in which the value and label of one claim are highly correlated with the value and label of other claims across multiple models.

Explanation. The goal of AllegatorTrack is not only to discovery true claims amongst multi-source, conflicting ones but also to provide explanations to the user. Once the truth labeling result is produced, the user can select any claim labeled as true or false and get corresponding statistical explanations about the trustworthiness scores of sources or about the confidence scores of claims selected by the user.

Allegation. To generate the minimal number of perturbations to inject into the original data set, AllegatorTrack first identifies the most influential claims that support the results of a truth discovery model selected by the user and it computes the minimal number of claims and fictive sources to add to change the considered results.

V. AllegatorTrack in Action

We will demonstrate the truth discovery main features of AllegatorTrack (B4 and F1-F3 modules illustrated in Figure 1) on three use case scenarios. The first use case is based on the the Book data set from [13], originally collected by [12] crawling abebooks.com. Its characteristics are given in Table 1. This use case is used in Figures 2 and 3 to show AllegatorTrack in Action. The second use case uses the

Use Case	ClaimId	DataItem Example	SourceId	Value Examples	Data set Characteristics
Book	C_1	"ISBN23606924:Authors"	`Htbook`	"Richard Johnsonbaugh, Marcus Schaefer"	879 sources – 1,263 objects
	C_2		`Sandy Chong`	"Marcus Schaefer, Richard Johnsonbaugh"	24,331 claims
	C_3		`textbookxdotcom`	"Richard Johnsonbaugh"	1 attribute: *Author name*
	C_4		`textbooksNow`	"Johnsonbaugh"	Data type: List of Strings
	C_5		`Limelight Bookshop`	"Johnsonbaugh, Richard"	Gold standard count: 100 objects
	C_6		`A1Books`	"Johnsonbaugh, Richard, Schaefer, Marcus"	
Biography	C_1	"George VI:Born"	`2654847`	"12/14/1895"	771,132 sources – 10,862,648 claims
	C_2		`2654847`	"12/14/1896"	3,783,555 data items
	C_3		`68.12.170.214`	"12/14/1895"	9 attributes
	C_4		`68.12.170.214`	"12/14/1896"	Data type: Strings, Date, Numerical
	C_5		`68.12.170.214`	"12/14/1896"	Gold standard count: 2,626 values
Population	C_1	"Atlanta, Georgia:Population2004"	`12.216.80.221`	"1425000"	4,264 sources – 41,196 objects
	C_2		`12.169.67.194`	"425000"	49,955 claims
	C_3		`12.169.67.194`	'1425000"	1 attribute: City Population per year
	C_4		`1130745: Brendan3`	"419122"	Data type: Number
	C_5		`131.95.178.163`	"1419122"	Gold standard count: 301 values
	C_6		`343214: Derek.cashman`	"425000"	

TABLE I: Claim examples and data set characteristics

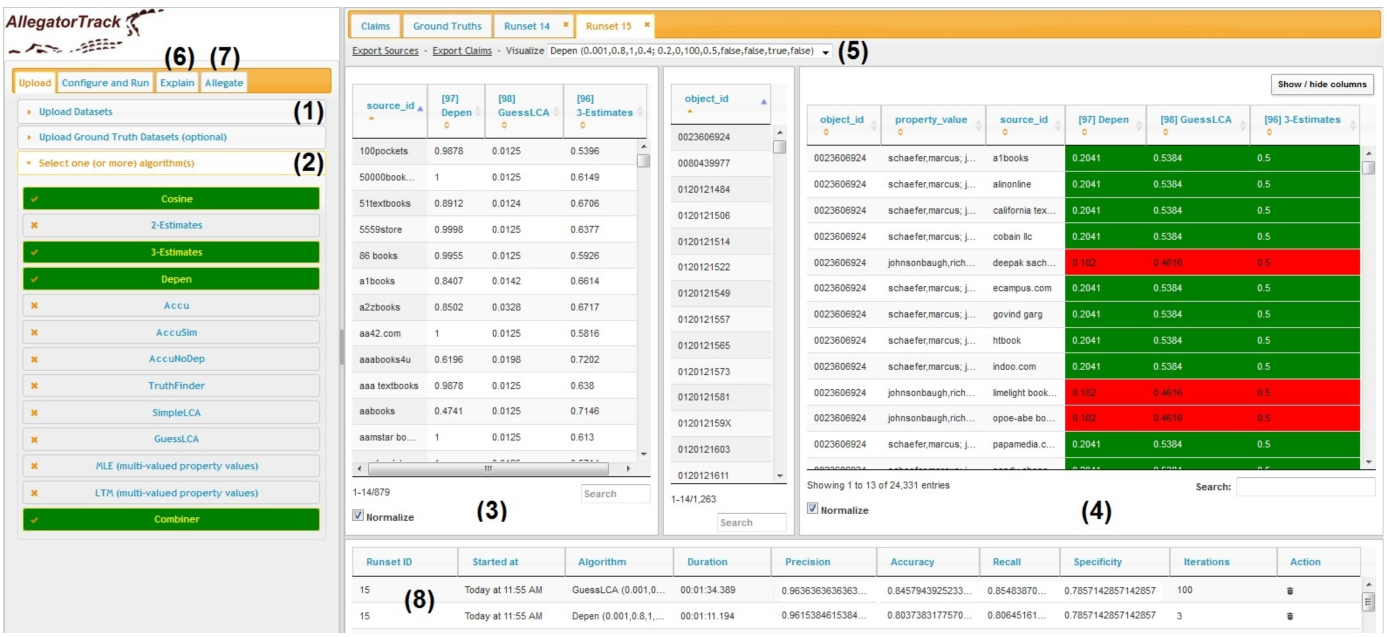

Fig. 2: AllegatorTrack in action over the AbeBooks.com data set

biography information collected on 1,863,248 people from 771,132 sources on the Web with in total 10,862,648 claims over 9 attributes (*Born, Died, Spouse, Father, Mother, Children, Country, Height, Weight*). The third use case scenario is based on the Population data set from [6] which consists of 49,955 claims extracted from Wikipedia edits from 4,264 sources.

Figure 2 shows a screenshot of AllegatorTrack. The first tab on the left allows the user to upload a data set (see (1) in Figure 2) and see all the claims from multiple sources. On the right panel, the uploaded data set is structured in a table with the claim identifier, property name (e.g., author name for the Book data set), the value and its respective source. AllegatorTrack also allows the user to upload a gold standard if available to compute the quality measures (precision, recall, accuracy, and specificity) of the algorithms. In (2), the user can select one or many algorithms for truth discovery computation and also get the results from the Bayesian combiner. After the setting of the parameters for the selected algorithms and execution (in the second tab entitled "Configure and Run"), the user can visualize and normalize the results of the runset (e.g., Runset 15 in the figure) in terms of the trustworthiness score of each source computed by each method in the panel (3) and the confidence score of each claimed value in the panel (4) with a green cell background when the value is considered to be true and red otherwise for each method. All algorithms with various parameter settings can be executed in parallel. The results can be visualized with Sankey diagrams in (5) such as the diagram given in Figure 3. It represents for each source on the left, how many claims are discovered to be true (or false) by a selected algorithm and for a certain number of conflicts. In Figure 3, Depen model discovers that, among the false claims, 29 of the claims have 5 conflicting values coming from the underlined sources. In tab (6), for a selected run of a considered algorithm with specific parameter setting, AllegatorTrack provides explanations enabling the

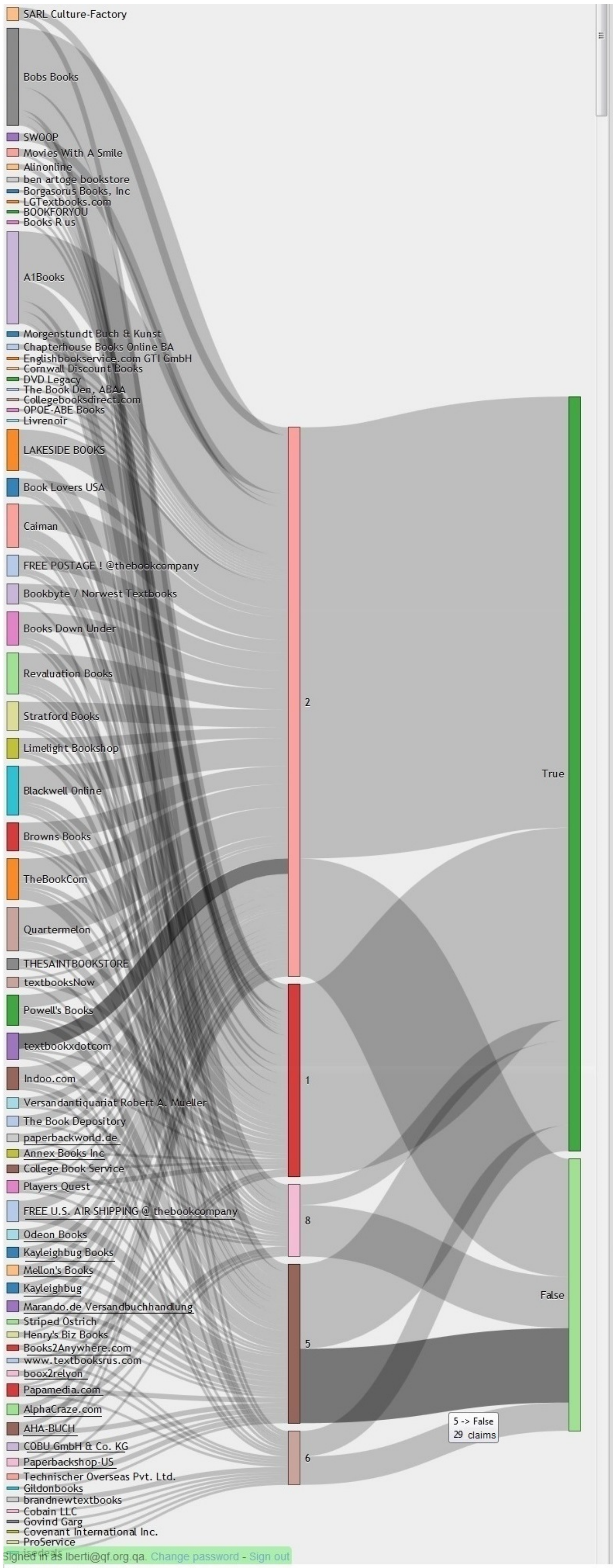

Fig. 3: AllegatorTrack Sankey diagram for Depen model applied to the Book data set

user to understand why a claimed value selected by the user is considered true (or false) by a given algorithm. Another interesting feature will be demonstrated related to collective inference of the truth discovery results from an ensemble of truth discovery methods. In tab (7), AllegatorTrack allows the users to generate the minimal set of allegations to change a specified output result either introducing a new source or adding claims to existing sources that will corroborate the user-defined allegations. Finally, a table of execution times and quality metrics representing the performance of the methods with respect to an uploaded ground truth data set is given in panel (8).

References

[1] X. Dong, L. Berti-Equille, Y. Hu, and D. Srivastava. SOLOMON: Seeking the Truth Via Copying Detection. *PVLDB*, 3(2):1617–1620, 2010.

[2] X. L. Dong, L. Berti-Equille, and D. Srivastava. Integrating conflicting data: The role of source dependence. *PVLDB*, 2(1):550–561, 2009.

[3] X. L. Dong, L. Berti-Equille, and D. Srivastava. Truth Discovery and Copying Detection in a Dynamic World. *PVLDB*, 2(1):562–573, 2009.

[4] A. Galland, S. Abiteboul, A. Marian, and P. Senellart. Corroborating Information from Disagreeing Views. In *WSDM*, pages 131–140, 2010.

[5] X. Li, X. L. Dong, K. Lyons, W. Meng, and D. Srivastava. Truth Finding on the Deep Web: Is the Problem Solved? *PVLDB*, 6(2):97–108, 2012.

[6] J. Pasternack and D. Roth. Knowing what to believe (when you already know something). In *COLING '10*, pages 877–885, 2010.

[7] J. Pasternack and D. Roth. Making better informed trust decisions with generalized fact-finding. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 2324–2329, 2011.

[8] J. Pasternack and D. Roth. Latent Credibility Analysis. In *WWW*, pages 1009–1020, 2013.

[9] D. A. Waguih and L. Berti-Equille. Truth Discovery Algorithms: An Experimental Evaluation, QCRI, Tech. Report, May 2014. .

[10] D. Wang, L. M. Kaplan, H. K. Le, and T. F. Abdelzaher. On Truth Discovery in Social Sensing: a Maximum Likelihood Estimation Approach. In *IPSN*, pages 233–244, 2012.

[11] Y. Wu, P. K. Agarwal, C. Li, J. Yang, and C. Yu. Toward computational fact-checking. *PVLDB*, 7(7):589–600, 2014.

[12] X. Yin, J. Han, and P. S. Yu. Truth Discovery with Multiple Conflicting Information Providers on the Web. *IEEE Trans. Knowl. Data Eng.*, 20(6):796–808, 2008.

[13] B. Zhao, B. I. P. Rubinstein, J. Gemmell, and J. Han. A Bayesian Approach to Discovering Truth from Conflicting Sources for Data Integration. *PVLDB*, 5(6):550–561, 2012.