FactCatch: Incremental Pay-as-You-Go Fact Checking with Minimal User Effort

Preprint · May 2020

DOI: 10.13140/RG.2.2.27201.58722

6 authors, including:

Tam Nguyen
École Polytechnique Fédérale de Lausanne
66 PUBLICATIONS **613** CITATIONS
SEE PROFILE

Matthias Weidlich
Humboldt-Universität zu Berlin
222 PUBLICATIONS **5,411** CITATIONS
SEE PROFILE

Hongzhi Yin
The University of Queensland
199 PUBLICATIONS **4,233** CITATIONS
SEE PROFILE

Bolong Zheng
The University of Queensland
70 PUBLICATIONS **764** CITATIONS
SEE PROFILE

FactCatch: Incremental Pay-as-You-Go Fact Checking with Minimal User Effort

Nguyen Thanh Tam
École Polytechnique Fédérale de Lausanne

Matthias Weidlich
Humboldt-Universität zu Berlin

Hongzhi Yin
The University of Queensland

Bolong Zheng
Huazhong University of Science & Technology

Nguyen Quang Huy
Hanoi University of Science & Technology

Nguyen Quoc Viet Hung
Griffith University

ABSTRACT

The open nature of the Web enables users to produce and propagate any content without authentication, which has been exploited to spread thousands of unverified claims via millions of online documents. Maintenance of credible knowledge bases thus has to rely on fact checking that constructs a trusted set of facts through credibility assessment. Due to an inherent lack of ground truth information and language ambiguity, fact checking cannot be done in a purely automated manner without compromising accuracy. However, state-of-the-art fact checking services, rely mostly on human validation, which is costly, slow, and non-transparent. This paper presents FactCatch, a human-in-the-loop system to guide users in fact checking that aims at minimisation of the invested effort. It supports incremental quality estimation, mistake mitigation, and pay-as-you-go instantiation of a high-quality fact database.

1 INTRODUCTION

Modern society faces an unprecedented amount of unverified claims, which do harm to democracy, economics, and national security [13]. While up to three quarters of facts posted by Web sources are false [17], journalism and politics have been impacted by 'false facts' on a global scale [12]. Building an open, yet trustworthy Web requires fact checking, i.e., the assessment of the credibility of emerging claims towards a fact database [7, 10].

Credibility assessment can use automated classification methods [15].While these methods scale to the volume of Web data, they are hampered by the inherent ambiguity of natural language, deliberate deception, and domain-specific semantics [16]. Hence, algorithms often fail to decipher complex contexts of claims [3]. Automatic methods further require large amounts of curated data, which is typically not available since such data quickly becomes outdated. Moreover, having algorithmic models judge the truth of claims raised ethical concerns on fairness and transparency [9, 14].

Against this background, several state-of-the-art fact checking services such as Snopes, PolitiFact, and FactCheck, rely on human feedback to validate claims [1]. However, eliciting user input is challenging. User input is expensive, in terms of time and cost. Hence, a timely validation of controversial claims quickly becomes infeasible, even if one relies on a large number of users and ignores the overhead to achieve consensus among them. Also, claims published on the Web are typically not independent and any user-based assessment of their credibility shall be propagated between correlated claims. Finally, user input is commonly limited by some effort budget, which bounds the number of claims to be validated.

This demo presents FactCatch,[1] a system for inCremental pAy-as-you-go facT CHecking with minimal user effort. The system's contributions are summarized as follows:

(1) *Incremental quality estimation:* FactCatch features an efficient probabilistic model to reason on the credibility of claims. It exploits mutual reinforcing relations between Web sources and claims to assess the credibility of unchecked claims.
(2) *Effort minimisation:* FactCatch guides a user in the fact checking process, while reducing the amount of validation effort needed to achieve a specific level of result precision.
(3) *Mistake mitigation:* FactCatch helps to identify suspicious user input; claims that may have been validated by mistake.
(4) *Pay-as-you-go instantiation:* FactCatch supports the separation of credible and non-credible claims at any time, to serve downstream applications with a high-quality fact database.
(5) *Early termination:* FactCatch includes means to stop fact checking to avoid to spend effort on marginal improvements of the quality of the fact database.

FactCatch is one of the principal systems [2] for guided fact checking that combines the best of automatic and manual approaches: Users validate the results of algorithmic models, while they save efforts by validating solely the claims that are most-beneficial for credibility assessment. FactCatch further enhances the explainability of algorithmic models by showing how the credibility of a claim is derived from its relation to trustworthy sources.

2 USER INTERACTION

2.1 Requirements

To incorporate users as a first-class citizen in fact checking, several requirements need to be met:

(R1) *Guided:* Fact checking shall be guided rather than an ad-hoc process of collecting data sources and arguments.
(R2) *Incremental:* Fact checking should exploit results of prior validations for reasons of efficiency and collaboration: Validations of one user are reused by another one.
(R3) *Pay-as-you-go:* Fact checking should continuously improve the credibility assessment, while enabling the instantiation of a fact database at any time. Users may then examine the database to decide whether to stop or resume validation.

Conference'17, July 2017, Washington, DC, USA
2020. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

[1] https://factcatch.github.io/
[2] https://fullfact.org/automated

2.2 The FactCatch Process

FactCatch is the first system that aims to address the above requirements, while minimizing user efforts. Fig. 1 illustrates the process behind FactCatch, which starts with a claim database that contains claims (candidate facts) extracted from the Web. The system adopts a human-in-the-loop interaction scheme, where the claim database is continuously updated in a pay-as-you-go manner, by:

(1) *selecting* a claim for which user feedback shall be sought;
(2) *eliciting* user input on the credibility of the selected claim, which either confirms it as credible or labels it as non-credible;
(3) *inferring* the credibility of remaining claims upon user input by an algorithmic model;
(4) *instantiating* the grounding that captures the facts that are assumed to be credible. At any time, a trusted set of facts can be used for downstream applications.

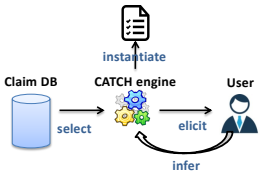

Figure 1: Overview of the FactCatch process

In the above process, steps (1), (3), and (4) need to be instantiated with specific methods, which will be described in §3.

2.3 Validation Dashboard

In FactCatch, user interaction is enabled through a rich validation dashboard. As shown in Fig. 2, it consists of multiple interactive views on top of a given claim database.

Landscape View. This view gives an overview of claims in a two-dimensional canvas, in which rows are data sources and columns are claims. Each source has a list of claims that it provides. The canvas includes functionality to filter data sources by name. When a user hovers a row of the canvas, an info pane is offered to show the trustworthiness and the number of claims of the respective source. Clicking on a cell, the dashboard switches to the Claim View. It also offers a matrix layout where user can reorder columns and rows to identify co-clusters of sources and claims.

Claim View. This view allows the user to investigate a particular claim and decide on its credibility. The default meta-data tab explains the characteristics of a claim, such as origin, description, example Web document or social post, original publication date, and last-mention date. The second tab is a graph representation of the data sources mentioning this claim and their further claims. Such representation enables the user to investigate the relations between different claims to decide on their credibility. The view also includes a navigation pane to search and rank the claims. The ranking function minimises user efforts by guiding user to focus on the most beneficial claims first, which is explained in §3.

Metric View. This view summarises the fact checking process by providing quality indicators such as uncertainty measures and the number of unverified claims. It also includes a chart panel that depicts the histogram of claim credibilities, which supports an assessment of the convergence of the validation process, see §3.

3 SYSTEM COMPONENTS

3.1 Functionality

Quality estimation by a probabilistic graphical model. A key metric is to determine the current quality of the fact database, which we formulate as a joint probability computation over a network of claims, sources, and Web documents. Here, the network structure is derived from the fact that a claim can be involved in multiple documents, each being provided by a different source.

To model these complex relations, and eventually derive the assignment of credibility probabilities, we rely on a Conditional Random Field (CRF). It is constructed as an undirected graph of three sets of random variables for sources, documents, and claims. Direct relations are captured by relation factors in the CRF, also called cliques since they always involve three random variables (source, document, claim). As a result, the *infer* function can be implemented by belief propagation over the CRF [10].

This model has several advantages for guided fact checking: (1) Each claim is assigned a correctness probability to model its credibility. This yields better interpretability than binary labels assigned by common classification models [10]. (2) The trustworthiness of each data source can be quantified as the probability of its respective random variable, supporting validation decisions during the fact checking process. (3) All user inputs obtained up to a certain point are incorporated easily via belief propagation, whereas other models face non-convex optimisation of online learning.

Effort minimisation by information gain theory. There is a trade-off between the precision of a knowledge base (the ratio of credible facts) and the amount of user input: The more claims are checked manually, the higher the precision. Since a validation of all claims is infeasible, FactCatch provides guidance by selecting and ranking claims for which user feedback should be sought based on the expected benefit. The latter is measured for a claim using an information theoretic model, which leverages the aforementioned probability information. Then, the claim with the highest information gain is shown first for user feedback elicitation.

More precisely, we define a conditional variant of the entropy over credibility values. It measures the expected entropy of the database under a specific validation input. The expected difference in uncertainty before and after incorporating input for a claim is the respective change in entropy (i.e. information gain). It quantifies the potential benefit of knowing the true value of a claim [10].

Mistake mitigation by indicative indexes. When validating claims, a user may make mistakes such as accidental confirmations of a (wrong) inferred credibility value of a claim. To mitigate such issues, we provide several indicative indexes:

- *Support information:* A user is confronted with the current inferred credibility of the claim to validate, along with a trustworthiness assessment of related sources. Any decision to deviate from the current most likely credibility assignment is typically taken well-motivated.
- *Redo feature:* A user can undo or change a previous validation. Belief propagation enabled by our probabilistic graphical model allows such anytime modification without recomputing the probabilities from scratch.

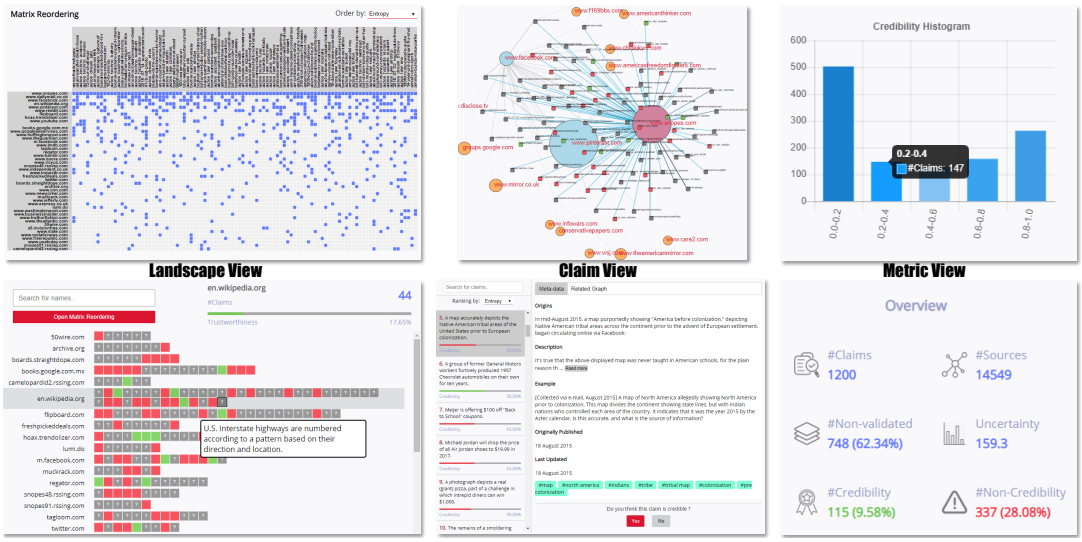

Figure 2: Multi-view Validation Dashboard

- *Confirmation check:* We also developed a lightweight periodic check that is triggered after a fixed number of iterations of the validation process. For a random claim that has been validated, we compute a snapshot of the probability model while leaving out the validation of that claim. Then, the probability of the claim is compared with the respective user input. If there is a contradiction, then the claim is identified as a potential mistake and the user is notified.

Pay-as-you-go approach by view maintenance. FactCatch enables the construction of a trusted set of facts from the claim database at any time. Technically, a grounding that decides which claims are deemed credible is instantiated every time the user input of an iteration of the validation process has been incorporated. This is done by taking the configuration with maximal joint probability on top of the probabilistic model. However, solving the maximal configuration is similar to solving a Boolean satisfiability problem. Thus, we leverage the most recent Gibbs sampling result of the expectation-maximization algorithm obtained during model inference, for instantiation [10]. Note that our system also supports the instantiation of the most non-credible claims (i.e., fake news).

Early termination by convergence indicators. FactCatch further reduces validation effort by observing the convergence of the probabilistic model. This is due to the property of diminishing returns of the information gain metric, where further user input leads to smaller improvements of the grounding quality. Our model enables the detection of such scenarios by several indicators.

- *Uncertainty reduction rate:* The reduction of uncertainty of the claim database is monitored over time.
- *Grounding change:* The number of claims for which the credibility is switched after each user validation is counted.
- *Human-algorithmic consensus:* measures the mismatch between user input and model grounding by KL divergence.
- *Precision improvement rate:* The precision of the model grounding is estimated based on k-fold cross validation.

3.2 Scalability

Incremental inference. We developed a novel algorithm for incremental inference, *iCRF*, which adopts the view maintenance principle by keeping a set of Gibbs samples over time [10]. Estimation of credibility and model parameters then exploits the results of the previous iteration of the validation process.

Batch validation. Users who validate claims face significant set-up costs, e.g., to familiarise with a particular domain. It therefore increases user convenience and efficiency if validation happens in batches of claims. We support such batching by selecting a subset of k claims with highest *joint* information gain. While the respective subset maximisation problem is NP-complete, we approximate it by greedy search based on its submodularity property [10].

Streaming fact checking. In practice, claims may arrive continuously from Web sources (e.g., social media). However, the model shall not be recomputed from scatch whenever a new claim arrives.

Upon the arrival of new sources and claims, the model structure and its parameters need to be updated. We therefore develop an online expectation-maximization algorithm that reuses and updates the previous trained parameters, which accelerates convergence in the presence of new data. The core of this algorithm is the stochastic approximation for the expectation step and L2-regularised Trust Region Newton Method for the maximization step [10].

3.3 Implementation

To enable cross-platform deployment and potential collaboration between validating users, the system is implemented as a web application. The claim database is stored in PostgreSQL. The back-end is written using Flask framework. The front-end is an HTML5 website communicating with the server using Python. The system follows the Model-View-Controller pattern which includes a service layer to handle the main fact checking process.

4 DEMONSTRATION PLAN

Datasets. The demo uses state-of-the-art datasets for fact checking systems:

- *Snopes* stems from the by-far most reliable and largest platform for fact checking [13]. It covers domains such as news and social media, and includes 14549 sources and 1200 claims.
- *Wikipedia* contains proven hoaxes and fictitious people from Wikipedia with 1955 sources and 157 claims. The dataset was derived by using curated claims from Wikipedia as queries for a search engine to collect related Web pages.

Scenarios. A live demonstration will focus on three scenarios:

- *Single-user validation:* A user performs fact checking from a fresh dataset. The user chooses a data file to import into the database via the dashboard and subsequently validates claims in the Claim View. During the validation, the user may consult the Metric View to decide whether to continue or stop the process. The user may also export a trusted set of facts for reporting purposes.
- *Multiple-user validation:* A user resumes or continues an existing validation task. Instead of importing a data file, the user starts with the dashboard directly as the previous validation results are loaded from a PostgreSQL database. The Landscape View enables the user to see which claims have been validated and to understand the trustworthiness of data sources. Using the two-dimensional canvas, the user can then focus on a problematic cluster of sources and claims.
- *Reference validation:* FactCatch also stores labelled data from the literature for demonstration purposes. This enables the user to compare their validation with labels of fact checking experts (e.g., from Snopes.com) to gain deeper insights.

Demo Takeaways. FactCatch serves a diverse audience: It (i) helps *media organisations* to monitor their digital content and ensure information quality; (ii) helps *governments* to investigate social biases and national election interference; and (iii) helps the *public* to distinguish between legitimate news and deceptive content; The audience would enjoy highlighted benefits using our system:

- *Low waiting time:* of system update upon each validation [5].
- *Minimal effort:* obtain a fact database of high quality with significantly less effort than baseline guiding techniques [10].

5 RELATED WORK

Extracting factual knowledge from Web data plays an important role in the construction of knowledge bases (KB) such as Freebase, YAGO and DBpedia, which store millions of facts about various domains [8].Some systems focus on extracting candidate facts (aka claims) [4], but leave the credibility assessment for expert services [1]. Other systems focus on static classification of claims, but neglect the dynamic and evolving nature of the Web [15, 16].

Going beyond the state-of-the-art, FactCatch provides a pay-as-you-go system [2, 6, 11], which can be run autonomously or on top of existing claim extraction frameworks and expert services. It supports an incremental guided fact checking process with minimal effort and complete control over the quality and transparency.

6 CONCLUSION

This paper presented FactCatch, a system to overcome the limitations of existing methods for automatic and manual fact checking. The system is not limited to experts, but enables any user to participate in incremental, pay-as-you-go fact checking in a transparent and guided manner. Highlights of FactCatch are: (i) Claims are not analysed individually but their complex network structure through documents and data sources is incorporated; (ii) claims are automatically ranked for validation to minimise user efforts; (iii) a single dashboard gives full control over the fact checking process; and (iv) a trusted set of facts may be instantiated at any time in the process. The system is further optimized for scalability through methods for early termination, batching validation, and online learning. In future work, we intend to extend FactCatch with crowdsourcing functionality. By relying on mass fact checking, controlled by a cost-profit model that prioritises highly contagious rumours, we strive for timely damage mitigation of emerging false claims.

REFERENCES

[1] Petter Bae Brandtzaeg, Asbjørn Følstad, and Maria Ángeles Chaparro Domínguez. 2018. How journalists and social media users perceive online fact-checking and verification services. *JP* 12, 9 (2018), 1109–1129.

[2] Phan Thanh Cong, Nguyen Thanh Tam, Hongzhi Yin, Bolong Zheng, Bela Stantic, and Nguyen Quoc Viet Hung. 2019. Efficient User Guidance for Validating Participatory Sensing Data. *ACM Transactions on Intelligent Systems and Technology (TIST)* 10, 4 (2019), 1–30.

[3] Chi Thang Duong, Quoc Viet Hung Nguyen, Sen Wang, and Bela Stantic. 2017. Provenance-Based Rumor Detection. In *ADC*. 125–137.

[4] Naeemul Hassan, Gensheng Zhang, Fatma Arslan, Josue Caraballo, Damian Jimenez, Siddhant Gawsane, Shohedul Hasan, Minumol Joseph, Aaditya Kulkarni, Anil Kumar Nayak, et al. 2017. ClaimBuster: the first-ever end-to-end fact-checking system. *PVLDB* 10, 12 (2017), 1945–1948.

[5] Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Vinh Tuan Chau, Tri Kurniawan Wijaya, Zoltán Miklós, Karl Aberer, Avigdor Gal, and Matthias Weidlich. 2015. SMART: A tool for analyzing and reconciling schema matching networks. In *ICDE*. 1488–1491.

[6] Nguyen Quoc Viet Hung, Duong Chi Thang, Matthias Weidlich, and Karl Aberer. 2015. Minimizing efforts in validating crowd answers. In *SIGMOD*. 999–1014.

[7] Nguyen Quoc Viet Hung, Huynh Huu Viet, Nguyen Thanh Tam, Matthias Weidlich, Hongzhi Yin, and Xiaofang Zhou. 2017. Computing crowd consensus with partial agreement. *IEEE Transactions on Knowledge and Data Engineering* 30, 1 (2017), 1–14.

[8] Nguyen Quoc Viet Hung, Kai Zheng, Matthias Weidlich, Bolong Zheng, Hongzhi Yin, Nguyen Thanh Tam, and Bela Stantic. 2018. What-if Analysis with Conflicting Goals: Recommending Data Ranges for Exploration. In *ICDE*. 1–12.

[9] Quoc Viet Hung Nguyen, Thanh Tam Nguyen, Zoltán Miklós, Karl Aberer, Avigdor Gal, and Matthias Weidlich. 2014. Pay-as-you-go reconciliation in schema matching networks. In *ICDE*. 220–231.

[10] Thanh Tam Nguyen, Matthias Weidlich, Hongzhi Yin, Bolong Zheng, Quoc Viet Hung Nguyen, and Bela Stantic. 2019. User guidance for efficient fact checking. *PVLDB* 12, 8 (2019), 850–863.

[11] Nguyen Quoc Viet Hung, Duong Chi Thang, Matthias Weidlich, and Karl Aberer. 2015. Erica: Expert guidance in validating crowd answers. In *SIGIR*. 1037–1038.

[12] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. In *KDD*, Vol. 19. 22–36.

[13] Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science* 359, 6380 (2018), 1146–1151.

[14] Weiqing Wang, Hongzhi Yin, Zi Huang, Qinyong Wang, Xingzhong Du, and Quoc Viet Hung Nguyen. 2018. Streaming Ranking Based Recommender Systems. In *SIGIR*. 525–534.

[15] You Wu, Pankaj K Agarwal, Chengkai Li, Jun Yang, and Cong Yu. 2017. Computational fact checking through query perturbations. *TODS* 42, 1 (2017), 1–41.

[16] Gensheng Zhang and Chengkai Li. 2018. Maverick: a system for discovering exceptional facts from knowledge graphs. *PVLDB* 11, 12 (2018), 1934–1937.

[17] Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. 2018. Detection and resolution of rumours in social media: A survey. *CSUR* 51, 2 (2018), 1–36.