Initial Plan: Implementation and Analysis of Truth-Discovery Algorithms

CM3203 One Semester Individual Project – 40 credits

Joe Singleton (student) Richard Booth (supervisor)

January 2019

1 Project Description

Truth-discovery algorithms are algorithms that aim to return the true state of the world (the 'facts') given an input consisting of various, possibly conflicting, reports from different sources of unknown trustworthiness and reliability. A main characteristic of these algorithms is that they are able to infer – from the input alone – not only belief values associated to different statements (reflecting the likelihood that they correspond to the truth), but also some measure of trust over the various sources. Moreover each of these two different kinds of values should cohere with each other, so that a statement receives a high belief value if it is backed up by highly trustworthy sources, while a source receives a high trust value if it provides highly believable statements.

The design of such algorithms has received an increasing amount of attention in recent years, especially with regard to aggregation of information on the web. However the emphasis has been on practical aspects (speed, efficiency etc. . .) rather than theoretical foundations. Furthermore the whole process is usually regarded as a 'one-shot' affair, in which it is assumed that all relevant information has been provided upfront.

This project will have both practical and theoretical components. On the practical side, a selection of truth-discovery algorithms from the literature will be implemented.

The implementation will provide a uniform interface for users to run various truth-discovery algorithms on their own data and evaluate the performance of each algorithm, both in terms of efficiency and how well the true 'facts' are discovered (in cases where truth or reliability values are already known).

For the theoretical component, some basic axioms of truth-discovery algorithms will be identified, and the implemented algorithms will be compared against each other with regard to whether these properties are satisfied.

For example, a basic principle of truth-discovery algorithms is that a source with highly believable statements receives a high trust value and vice versa; in this project we will attempt to express this principle in a more precise way as an axiom of truth-discovery.

Other axioms may concern the changes in trust/belief scores resulting from small changes the source/claim network.

It may additionally be possible find a sound and complete axiomatisation of a particular algorithm; i.e. an algorithm X could be characterised by a list of axioms such that any algorithm Y satisfies these axioms if and only if $X = Y$. This has been done for the *PageRank* algorithm for the ranking of web pages in [1].

2 Project Aims and Objectives

In this project I aim to:

- Implement between three and six truth-discovery algorithms in Python. Command line, Python API and web-based interfaces (time permitting) will be provided. The user will be able to provide their own input data and select their desired output format. Well known algorithms that could be implemented include *Hubs and Authorities* [3] (adapted to truth-discovery in [4]), *Average·Log*, *Investment*, *PooledInvestment* [4], *TruthFinder* [5], *Cosine*, *2-Estimates* and *3-Estimates* [2].
- Allow users to provide data incrementally, and look at the fluctuation of the output belief and trust values when sources provide their information gradually over time instead of all in one go.
- Apply the implemented algorithms to synthetic and real-world datasets, and compare the performance of each against a baseline method (e.g. majority voting).
- Analyse the properties of existing algorithms by identifying some basic axioms of truth-discovery. Each of the implemented algorithms will then be compared with regard to whether they satisfy these axioms. For example, it would be interesting to identify an axiom for each algorithm that distinguishes it from the other algorithms. Additionally, the plausibility and intuition behind each axiom will be discussed.
- Focus specifically on the *Hubs and Authorities* [3, 4], algorithm, and compile a list of sound properties that is complete as possible, as time permits.

3 Work Plan

The main deliverables of this project are:

- A software implementation of a selection of truth-discovery algorithms, with documentation.
- Real-world and synthetic datasets applicable to truth-discovery, and a comparison of the performance of the implemented algorithms on each dataset.

- A list of five to ten axioms for truth-discovery, with discussion of their interpretation and plausibility.
- A final report with detailed discussion of the software implementation, the algorithms implemented, and the identified axioms.

Below is a list of milestones for progress towards delivering the above and achieving the aims of section 2, and dates by which I aim to complete these milestones.

Note that although there are dependencies between some of the deliverables listed above (e.g. the report cannot be written before the software implementation), the plan below assumes that work on certain independent tasks will overlap.

By 8th February (Friday week 2):

- A literature review has been carried out to identify which algorithms from the literature should be implemented. Enough background knowledge has been obtained to start working on the implementation.
- The forms of input/output data for the implementation has been decided on.

By 15th February (Friday week 3):

- Work on the Python implementation has started.

By 22nd February (Friday Week 4):

- First review meeting with supervisor held.
- Progress towards axioms: a common framework for truth-discovery has been set out, so that the developed axioms will be applicable to all algorithms.

By 8st March (Friday week 6):

- Base python implementation finished for all algorithms. It should be possible for a user to give their own input data, and receive output using a simple command-line interface or by using an API from Python code.
- One or two basic axioms for truth-discovery have been identified.

By 15st March (Friday week 7):

- Implementation supports viewing the fluctuation of output belief when providing data incrementally (this could be in the form of graphs or tables).

By 22nd March (Friday week 8):

- Work on a web interface for the Python implementation has started.

- An interesting real-world dataset has been identified, and a synthetic dataset produced. The implemented algorithms have been applied to each, and conclusions drawn regarding the performance of each algorithm in both real-world and synthetic cases.
- Second review meeting with supervisor held.

By 5th April (Friday week 10):

- Progress made on initial sections of final report, ahead of the Easter break.
- The web interface is complete and provides the same functionality as the command-line interface.
- Documentation for the implementation, including API usage, has been created.

By 12th April (Friday week 11):

- Five to ten axioms have been identified, and each algorithm compared with respect to whether they satisfy the axioms.

By 10th May (Friday week 12):

- Final report written and submitted.

References

[1] Alon Altman and Moshe Tennenholtz. Ranking systems: The pagerank axioms. In *Proceedings of the 6th ACM Conference on Electronic Commerce*, EC '05, pages 1–8, New York, NY, USA, 2005. ACM.

[2] Alban Galland, Serge Abiteboul, Amélie Marian, and Pierre Senellart. Corroborating information from disagreeing views. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, WSDM '10, pages 131–140, New York, NY, USA, 2010. ACM.

[3] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, September 1999.

[4] Jeff Pasternack and Dan Roth. Knowing what to believe (when you already know something). In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 877–885, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[5] X. Yin, J. Han, and P. S. Yu. Truth discovery with multiple conflicting information providers on the web. *IEEE Transactions on Knowledge and Data Engineering*, 20(6):796–808, June 2008.