A Precise Information Flow Measure from Imprecise Probabilities

Sari Haj Hussein

1Department of Computer Science
Aalborg University

2012-06-21
1 Introduction

2 Representing Agent’s Uncertainty

3 Capturing Belief

4 Arithmetic on Beliefs

5 Language & Lifted Language

6 Inference Scheme

7 Experimenting with Inference Scheme

8 Measuring Information Flow
Literature

- ...A Precise Information Flow Measure from Imprecise Probabilities
Literature


- ...A Precise Information Flow Measure from Imprecise Probabilities
Field

- Quantitative Information Flow (QIF) analysis
- Decide the number of bits that might be revealed from a program’s secret input during the execution of that program
Qualitative...
Problem

- The QIF metric by Clarkson et al
- It uses Bayesian inference
- It captures the improvement in the attacker’s belief as she interacts with a program’s execution
- Thereby it quantifies the flow
Contributions

- The paper presents a justified generalization of the analysis method done by Clarkson et al.
- It highlights the weaknesses in the original work.
- It shows that they are eliminated by way of the generalization.
- The generalization is based on one of the theories of imprecise probabilities, namely the theory of evidence.
Contributions

Dempster

Shafer
1 Introduction

2 Representing Agent's Uncertainty

3 Capturing Belief

4 Arithmetic on Beliefs

5 Language & Lifted Language

6 Inference Scheme

7 Experimenting with Inference Scheme

8 Measuring Information Flow
Frame of Discernment (Sample Space)

- A set of possible worlds that an agent considers possible
- \( \mathcal{W} = \{ \text{password, 123456, qwerty, abc123, letmein, 696969} \} \)

Closed-world Assumptions (Shafer’s Model)

- Exclusiveness: At most one of the worlds in \( \mathcal{W} \) is the true world
- Exhaustiveness: \( \mathcal{W} \) contains all the possible worlds
## Frame of Discernment

### Frame of Discernment (Sample Space)
- A set of possible **worlds** that an agent considers possible
- \( \mathcal{W} = \{ \text{password, 123456, qwerty, abc123, letmein, 696969} \} \)

### Closed-world Assumptions (Shafer’s Model)
- Exclusiveness: At most **one** of the worlds in \( \mathcal{W} \) is the true world
- Exhaustiveness: \( \mathcal{W} \) contains **all** the possible worlds
Frame of Discernment

Frame of Discernment (Sample Space)
- A set of possible worlds that an agent considers possible
- \( \mathcal{W} = \{ \text{password, 123456, qwerty, abc123, letmein, 696969} \} \)

Closed-world Assumption (Shafer’s Model)
- Exclusiveness: At most one of the worlds in \( \mathcal{W} \) is the true world
- Exhaustiveness: \( \mathcal{W} \) contains all the possible worlds
Frame of Discernment

Dezert

Smarandache
Frame of Discernment

Example

- \( PWC : \) if \( p = g \) then \( a := 1 \) else \( a := 0 \) \( p \in \{A, B, C\} \)
- \( r = \{p, g, a\}, h = \{p\}, l = \{g, a\} \)
- \( W_h = \prod_{x \in \{p\}} W_x = W_p = \{A, B, C\} \)
- \( W_l = \prod_{x \in \{g, a\}} W_x = W_g.W_a = \{A, B, C\}.\{0, 1\} \)
- \( W_{h\cup l} = \prod_{x \in \{p, g, a\}} W_x = W_p.W_g.W_a = \{A, B, C\}.\{0, 1\} = \{(A, A, 0), \ldots\} \)
Frame of Discernment

Example

- PWC: if \( p = g \) then \( a := 1 \) else \( a := 0 \) \( p \in \{A, B, C\} \)
- \( r = \{p, g, a\}, h = \{p\}, l = \{g, a\} \)
- \( \mathcal{W}_h = \prod_{X \in \{p\}} \mathcal{W}_X = \mathcal{W}_p = \{A, B, C\} \)
- \( \mathcal{W}_l = \prod_{X \in \{g, a\}} \mathcal{W}_X = \mathcal{W}_g.\mathcal{W}_a = \{A, B, C\}.\{0, 1\} \)
- \( \mathcal{W}_{h \cup l} = \prod_{X \in \{p, g, a\}} \mathcal{W}_X = \mathcal{W}_p.\mathcal{W}_g.\mathcal{W}_a = \{A, B, C\}.\{0, 1\} = \{(A, A, 0), ...\} \)
Belief Functions

- Frame of Discernment is too coarse
- Comparing the likelihood of worlds is not possible
- Belief functions is a numeric representation of uncertainty that enables full ordering of worlds
Belief Functions vs. Probability Measures

- The finite additivity property

\[ \text{Pro}(X_1 \cup \ldots \cup X_n) = \text{Pro}(X_1) + \ldots + \text{Pro}(X_n) \]

- You are forced to work with singleton sets
- No overlapping sets

\[ \text{Pro}(\{A, B\}) = 0.2, \quad \text{Pro}(\{B, C\}) = 0.3 \]

- No nested sets

\[ \text{Pro}(\{A, B\}) = 0.2, \quad \text{Pro}(\{A, B, C\}) = 0.3 \]
Belief Functions vs. Probability Measures

- Ignorance is difficult to represent

\[ \text{Pro}(\{A\}) = 0.2, \text{Pro}(\{A, B\}) = 0.0 \]

- Contradiction is difficult to represent

\[ \text{Pro}(\{A\}) = 0.2, \text{Pro}(\{B\}) = 0.3, \text{Pro}(\emptyset) = 0.5 \]

- Modeling collaboration is not possible
- Add the computational problem...
1. Introduction
2. Representing Agent's Uncertainty
3. Capturing Belief
4. Arithmetic on Beliefs
5. Language & Lifted Language
6. Inference Scheme
7. Experimenting with Inference Scheme
8. Measuring Information Flow
Mass Function

- $m: \mathcal{P}(\mathcal{W}_s) \rightarrow [0, 1]$
- $m(\emptyset) = 0, \sum_{A \in \mathcal{P}(\mathcal{W}_s)} m(A) = 1$
- $m(A)$ the degree of belief that the true world is in $A$

Belief Function

- $\text{Bel}: \mathcal{P}(\mathcal{W}_s) \rightarrow [0, 1]$
- $\text{Bel}(A) = \sum_{B \subseteq A} m(B)$
Mass Function

- $m : \mathcal{P}(\mathcal{W}_s) \rightarrow [0, 1]$
- $m(\emptyset) = 0, \sum_{A \in \mathcal{P}(\mathcal{W}_s)} m(A) = 1$
- $m(A)$ the degree of belief that the true world is in $A$

Belief Function

- $Bel : \mathcal{P}(\mathcal{W}_s) \rightarrow [0, 1]$
- $Bel(A) = \sum_{B \subseteq A} m(B)$
Belief Combination

Combining two pieces of evidence $m_1$ and $m_2$ from two independent sources

1. \[(m_1 \otimes m_2)(A) = k \cdot \sum_{B \cap C = A} m_1(B) \cdot m_2(C)\]

2. \[(m_1 \otimes m_2)(\emptyset) = 0, k^{-1} = \sum_{B \cap C \neq \emptyset} m_1(B) \cdot m_2(C)\]

3. \[(m_1 \otimes m_2)(A) = k \cdot \sum_{B \triangleleft C = A} m_1(B) \cdot m_2(C)\]

4. \[(m_1 \otimes m_2)(\emptyset) = 0, k^{-1} = \sum_{B \triangleleft C \neq \emptyset} m_1(B) \cdot m_2(C)\]
Belief Conditioning

Current agent’s belief is captured using $m$

A new piece of evidence that the true world is in $B$

Agent can do a knowledge update...

$$m_B(A) = \begin{cases} k \cdot \sum_{C \cap B = A} m(C) & \text{for } A \neq \emptyset \\ 0 & \text{for } A = \emptyset \end{cases}$$

$$k^{-1} = \sum_{C \cap B \neq \emptyset} m(C)$$
Belief Divergence

- We need to measure the divergence between 2 mass functions in an **information-theoretic** manner.
- There is **no** out-of-the-box information-theoretic divergence in the theory of evidence.
- Divergence measures, that are based on **geometrical interpretations** of mass functions, do **not** work.
- We should derive a suitable divergence measure. How?
  1. **Start** with a divergence measure in probability theory.
  2. **Re-write** this divergence in terms of information-theoretic functionals.
  3. **Generalize** these functionals into the theory of evidence.
Belief Divergence

- Kullback-Leibler \( KL(p_1, p_2) = \sum_{x \in X} p_1(x) \log \frac{p_1(x)}{p_2(x)} \)
- Jensen-Shannon \( JS(p_1, p_2) = 2S\left(\frac{p_1 + p_2}{2}\right) - S(p_1) - S(p_2) \)

Generalized Jensen-Shannon Divergence

- \( GJS(m_1, m_2) = 2GS\left(\frac{m_1 + m_2}{2}\right) - GS(m_1) - GS(m_2) \)
- \( GS(m) = AU(Bel) - GH(m) \)
Belief Divergence

- Kullback-Leibler $KL(p_1, p_2) = \sum_{x \in \mathcal{X}} p_1(x) \log \frac{p_1(x)}{p_2(x)}$
- Jensen-Shannon $JS(p_1, p_2) = 2S\left(\frac{p_1 + p_2}{2}\right) - S(p_1) - S(p_2)$

**Generalized Jensen-Shannon Divergence**

- $GJS(m_1, m_2) = 2GS\left(\frac{m_1 + m_2}{2}\right) - GS(m_1) - GS(m_2)$
- $GS(m) = AU(Bel) - GH(m)$
In the paper...

- Imperative while-language
- Lift the syntax and semantics of it
- We are able to write program source code in terms of mass functions
1. Introduction

2. Representing Agent’s Uncertainty

3. Capturing Belief

4. Arithmetic on Beliefs

5. Language & Lifted Language

6. Inference Scheme

7. Experimenting with Inference Scheme

8. Measuring Information Flow
In the paper...

- Start from an attacker’s model
- Show how an attacker updates her knowledge from interacting with a program execution
- The arithmetic toolbox on beliefs and the execution rules of commands in the lifted language are extensively used here
1. Introduction
2. Representing Agent’s Uncertainty
3. Capturing Belief
4. Arithmetic on Beliefs
5. Language & Lifted Language
6. Inference Scheme
7. Experimenting with Inference Scheme
8. Measuring Information Flow
In the paper...

- $\mathcal{PWC}$: if $p = g$ then $a := 1$ else $a := 0 \quad p \in \{A, B, C\}$

<table>
<thead>
<tr>
<th>$\mathcal{P}(\mathcal{W}_h)$</th>
<th>$m_{pre}$</th>
<th>$m'_{post}$</th>
<th>$m''_{post}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${A}$</td>
<td>.98</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>${B, C}$</td>
<td>.02</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
In the paper...

- $\mathcal{PW}C : \text{ if } p = g \text{ then } a := 1 \text{ else } a := 0 \quad p \in \{A, B, C\}$

**TABLE VII**

The Attacker’s Prebelief and Postbelief in Experiment 2

<table>
<thead>
<tr>
<th>$\mathcal{P}(W_h)$</th>
<th>$m_{pre}$</th>
<th>$m'_{post}$</th>
<th>$m''_{post}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${A, B}$</td>
<td>.98</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>${A, B, C}$</td>
<td>.02</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>${A}$</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>${B}$</td>
<td>0</td>
<td>0</td>
<td>.98</td>
</tr>
<tr>
<td>${B, C}$</td>
<td>0</td>
<td>0</td>
<td>.02</td>
</tr>
</tbody>
</table>
Introduction

Representing Agent’s Uncertainty

Capturing Belief

Arithmetic on Beliefs

Language & Lifted Language

Inference Scheme

Experimenting with Inference Scheme

Measuring Information Flow
Measuring Information Flow

- When beliefs are involved then flow is the **improvement in the accuracy** of an attacker’s belief

**Our Flow Measure**

- The accuracy of the attacker’s prebelief is \( GJS(m_{pre}, \dot{m}_h) \)
- The accuracy of the attacker’s postbelief is \( GJS(m_{post}, \dot{m}_h) \)

\[
Q = GJS(m_{pre}, \dot{m}_h) - GJS(m_{post}, \dot{m}_h) = 2GS\left(\frac{m_{pre} + \dot{m}_h}{2}\right) - 2GS\left(\frac{m_{post} + \dot{m}_h}{2}\right) - GS(m_{pre}) + GS(m_{post})
\]
Measuring Information Flow

- When beliefs are involved then flow is the improvement in the accuracy of an attacker’s belief

**Our Flow Measure**

- The accuracy of the attacker’s prebelief is $GJS(m_{pre}, \dot{m}_h)$
- The accuracy of the attacker’s postbelief is $GJS(m_{post}, \dot{m}_h)$

$$Q = GJS(m_{pre}, \dot{m}_h) - GJS(m_{post}, \dot{m}_h)$$

$$= 2GS\left(\frac{m_{pre} + \dot{m}_h}{2}\right) - 2GS\left(\frac{m_{post} + \dot{m}_h}{2}\right) - GS(m_{pre}) + GS(m_{post})$$
Sample flow calculations for the experiments

pyuds

A Python library for measuring uncertainty in Dempster-Shafer theory.

Description

pyuds is a Python library for measuring uncertainty in Dempster-Shafer theory of evidence. The functionals supported are Generalized Hartley (GH) uncertainty functional, Generalized Shannon (GS) uncertainty functional, and Aggregate Uncertainty (AU) functional. The library can be utilized either through its API, or through a user-friendly web interface.
In the paper...

- The measure has the bounds $\varrho_Q = [-\eta, \eta]$
- The space of the exhaustive search can be easily determined
Reflection and Future Work

- Probability theory has its base in set theory, but imprecise probabilities do **not**!
- The application of imprecise probabilities in fields **other** than QIF could be rewarding
- **Subjective logic** by Jøsang is good at trust modeling but does not work in QIF
- Set of **stronger properties** related to $KL$ and $JS$ whose proofs could be rewarding
- Could be interesting to do **simulation** using larger frames of passwords
- Could be interesting to look at **guesswork** in this setting
Introduction
Representing Agent's Uncertainty
Capturing Belief
Arithmetic on Beliefs
Language & Lifted Language
Inference Scheme
Experimenting with Inference Scheme
Measuring Information Flow
Reflection and Future Work
Thank You

Pouly
Thank You!