A General-Purpose Protocol for Multi-Agent based Explanations

Giovanni Ciatto[*] Matteo Magnini[*] Berk Buzcu[†]
Reyhan Aydoğan[†] Andrea Omicini[*]

[*]Department of Computer Science and Engineering (DISI)
Alma Mater Studiorum—Università di Bologna (UniBo)
via dell'Università 50, Cesena FC 47522, Italy

[†]Department of Computer Science, Özyeğin University (OZU)
Nisantepe Mah. Orman Sok. No:34-36 Alemdağ, Çekmeköy, Istanbul 34794, Türkiye

22nd International Conference on
Autonomous Agents and Multiagent Systems (AAMAS 2023)
London, 2023-05-29

Next in Line. . .

1. Motivation & Context
2. Protocol design
3. Towards a software technology
4. Conclusions & future works

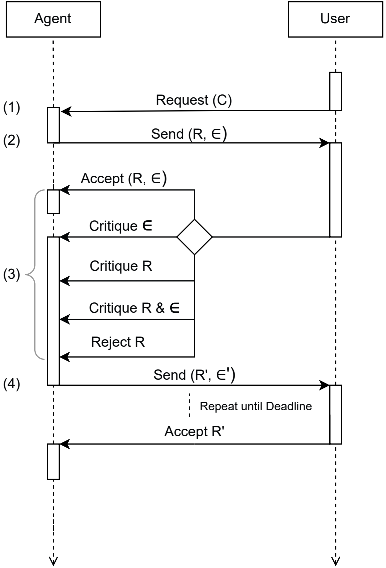
Context

- Pervasive exploitation of AI in modern recommender systems (RS)
  - query → predict → recommend
- Call for explainability in AI[Gunning, 2016]
  - need for AI systems to provide explanations of their decision-making processes
- Current research is about algorithms for *interpretability*
  - a.k.a. "opening the black box"[Guidotti et al., 2019]
  - major focus on *supervised* machine learning (ML) algorithms
- Interpretability vs. explainability[Ciatto et al., 2020]
  - interpretability is about easing humans understading
    - focus on representations
  - explainability is a dialogue between humans and machines
    - focus on interaction

Motivation

- Need for interactive explanations
  - explanations as dialogues among software and human agents
- Need for an abstract protocol for explanatory RS
  - fixing roles, dictating which messages to exchange, and when
- Need for a general-purpose software technology for that protocol

State-of-the-art protocols for explainable RS[Buzcu et al., 2022]



Lesson learnt:

- 2 roles
  - explainer / recommender / agent
  - explainee / user / human
- very abstract w.r.t. recommendation, explanation, critique
- multi-round request–reply protocol
- recommendation & explanation are coupled

Contribution of the paper

1. We specialize prior work[Buzcu et al., 2022] towards
   - on-demand explanations — users may just not need them some times
   - support for both motivational and contrastive explanations
     - e.g. “why X?” vs. “why X and not Y?”

2. We formalise a general-purpose protocol. . .

3. . . . and we design software technology for that protocol
   - interoperable with both ML and MAS technologies
   - supporting pluggability of recommendation/explanation strategies

Next in Line. . .

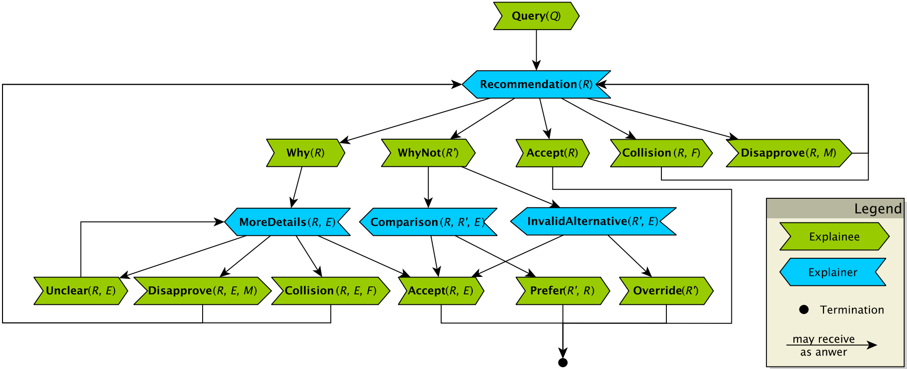
1 Motivation & Context

2 Protocol design

3 Towards a software technology

4 Conclusions & future works

Protocol overview I



Protocol overview II

Key features

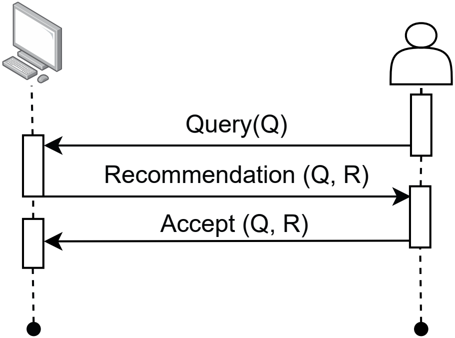
- request–response metaphor, initiated by explainee
- explanations are provided only upon request
- different workflows for motivational and contrastive explanations
- various sorts of critiques for each explanation type
- various sorts of acceptance/rejection situations for recommendations
- agnostic w.r.t. recommendation and explanation strategies/representations

Protocol overview III

Message payloads

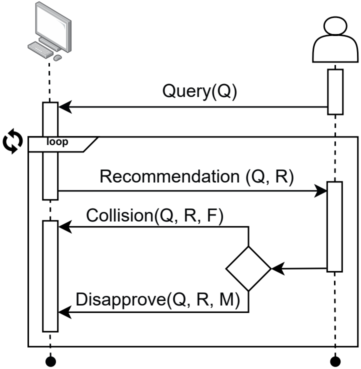
Queries	(Q) recommendation requests issued by the explainee
Recommendations	(R, R') responses to queries
Explanations	(E, E') information issued by the explainer to clarify recommendation;
Features	(F) justification for collision with explainee preference
Motivations	(M) justification for recommendation rejection

Protocol by examples I



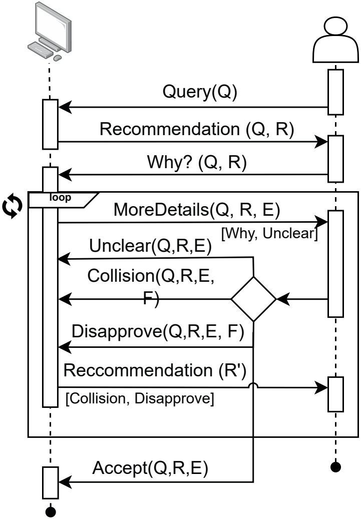
Quick accept: the user accepts the recommendation without asking for explanations

Protocol by examples II



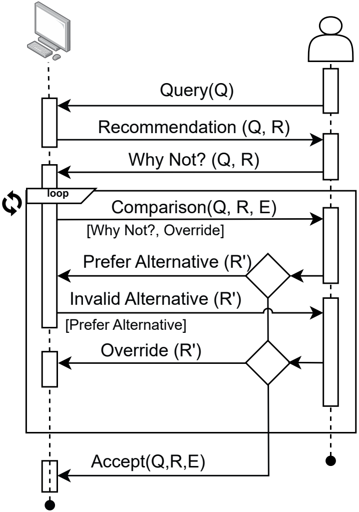
Quick retry: the user rejects the recommendation without asking for explanations. Another recommendation is proposed, accordingly.

Protocol by examples III



Ordinary explanation loop: the user asks 'why' after a recommendation, and then agent answers with further details. The request for details may be repeated several times.

Protocol by examples IV



Contrastive explanation loop: the user asks 'why not' another recommendation. The agent may then explain why the other recommendation is acceptable or invalid. The user may either accept the original recommendation or prefer their own.

Next in Line. . .

1 Motivation & Context

2 Protocol design

3 Towards a software technology

4 Conclusions & future works

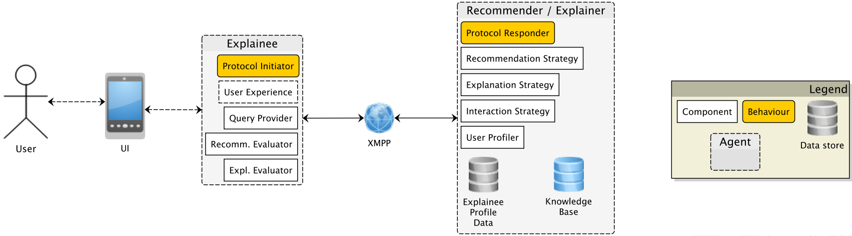
Requirements

- Compatibility with state-of-the-art ML technologies
  → target the Python technology
- Compatibility with state-of-the-art MAS technologies
  → implement the protocol in Spade*
- Pluggability of recommendation and explanation strategies
  → avoid hard-coding them, and provide a flexible API

*https://spade-mas.readthedocs.io

About PyXMas I

- Modular Python library providing a Spade-based implementation
  - WIP: `https://github.com/pikalab-unibo/pyxmas`
- Modules allow for pluggability of strategies



- Predefined Spade behaviours with callbacks for plugging strategies

About PyXMas II

About **explainer**-side modules

Recommendation Strategy: computes recommendations for any given query

Explanation Strategy: computes explanations for any given recommendation

User Profiler: learns user profiles from users' feedback

Interaction Strategy: decides how to present recommendations/explanations

About PyXMas III

About **explainee**-side modules

Query Provider: generates/prompt for queries

Recommendation Evaluator: decides whether to accept or reject recommendations

Explanation Evaluator: decides whether to accept or reject explanations

User Interface: necessary when the explainee is a human

Next in Line...

1. Motivation & Context
2. Protocol design
3. Towards a software technology
4. Conclusions & future works

Conclusions & future works

Summary of contributions

- Abstract recommendation + explanation protocol
  - supporting on-demand and contrastive explanations
- design of software technology implementing it
  - in a re-usable whay

Future works

- Complete PyXMas implementation
- Experiment with different strategies
- Evaluate the protocol with human subjects

A General-Purpose Protocol for Multi-Agent based Explanations

Giovanni Ciatto[*] Matteo Magnini[*] Berk Buzcu[†]
Reyhan Aydoğan[†] Andrea Omicini[*]

[*]Department of Computer Science and Engineering (DISI)
Alma Mater Studiorum—Università di Bologna (UniBo)
via dell'Università 50, Cesena FC 47522, Italy

[†]Department of Computer Science, Özyeğin University (OZU)
Nisantepe Mah. Orman Sok. No:34-36 Alemdağ, Çekmeköy, Istanbul 34794, Türkiye

22nd International Conference on
Autonomous Agents and Multiagent Systems (AAMAS 2023)
London, 2023-05-29

References I

[Buzcu et al., 2022] Buzcu, B., Varadhajaran, V., Tchappi, I., Najjar, A., Calvaresi, D., and Aydogan, R. (2022).

Explanation-based negotiation protocol for nutrition virtual coaching.

In Aydogan, R., Criado, N., Lang, J., Sánchez-Anguix, V., and Serramia, M., editors, *PRIMA 2022: Principles and Practice of Multi-Agent Systems - 24th International Conference, Valencia, Spain, November 16-18, 2022, Proceedings*, volume 13753 of *Lecture Notes in Computer Science*, pages 20–36. Springer

DOI:10.1007/978-3-031-21203-1_2.

References II

[Ciatto et al., 2020] Ciatto, G., Schumacher, M. I., Omicini, A., and Calvaresi, D. (2020).

Agent-based explanations in ai: Towards an abstract framework.

In Calvaresi, D., Najjar, A., Winikoff, M., and Främling, K., editors, *Explainable, Transparent Autonomous Agents and Multi-Agent Systems*, volume 12175 of *Lecture Notes in Computer Science*, pages 3–20. Springer, Cham.

Second International Workshop, EXTRAAMAS 2020, Auckland, New Zealand, May 9–13, 2020, Revised Selected Papers

DOI:10.1007/978-3-030-51924-7_1.

[Guidotti et al., 2019] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. (2019).

A survey of methods for explaining black box models.

ACM Comput. Surv., 51(5):93:1–93:42

DOI:10.1145/3236009.

References III

[Gunning, 2016] Gunning, D. (2016).
Explainable artificial intelligence (XAI).
Funding Program DARPA-BAA-16-53, DARPA
http://www.darpa.mil/program/explainable-artificial-intelligence.