https://dl.acm.org/doi/10.1007/978-3-030-86967-0_16

Aggregation for flexible Challenge-Response[1]

Samuel Brezani[a] and Peter Vojtas[a,b] [0000-0002-3526-8475]

[a]Globesy Ltd, Framborska 58, Zilina, Slovakia
[b]Dpt. Software Engineering, Charles University, Prague, Czechia
pvojtas@gmail.com

Abstract. A real problem use-case represents a challenge. This is usually transformed (reduced) to a model. We expect the model to give a response/solution which is (at least in a degree) acceptable/meets the challenge. Moreover, this challenge-response understanding has two levels – both the real world situation and model situation contains the challenge side (input, query, problem ...) and the response side (output, answer, solution ...). We present a formal model of ChRF-Challenge-Response Framework inspired by our previous work on Galois-Tukey connections. Nevertheless, real-world reduction to models needs some adaptation of this formal model. In this paper, we introduce several examples extending ChRF. We illustrate this using several practical situations, mainly in the area of recommender systems. Data of the model situations are motivated by Fagin-Lotem-Naor's data model with attribute preferences and multicriterial aggregation. In this realm, we review our previous work on the preferential interpretation of fuzzy sets, implicit behavior in/and online/offline evaluation of recommender systems. We finish with smart extensions of industrial processes. We propose a synthesis of these and formulate some open problems.

Keywords: Galois-Tukey Connect, Blass-Query-Answer, Challenge-Response Framework, Recommendation, Aggregation

1 Introduction

A real problem use-case represents a challenge. This is usually transformed (reduced) to a model. We expect the model to give a response/solution which is (at least in a degree) acceptable/meets the challenge. Moreover, this challenge-response understanding has two levels – both the real world situation and model situation contains the challenge side (input, query, problem...) and the response side (output, answer, solution...).

We can see these phenomena in many situations. For instance, when a declarative formulation needs a procedural implementation (SQL, NLP...). Or, a human perception

[1] This publication was realized with the support of the Slovak Operational Programme Integrated Infrastructure in the frame of the project: Intelligent systems for UAV real-time operation and data processing, code ITMS2014+: 313011V422 and co-financed by the European Regional Development Fund.

expects a relevant response (well, this can be difficult even between humans). Or, when a client requests some computation from a server... We consider this *"Challenge-Response in real-to-model reduction"* principle important and would like to develop it further.

These were already formally modeled by our previous work on the ChRF-Challenge-Response Framework with situations and reductions (see {KV ComSIS}, {VV-CLMPST}). The requirement was that the acceptable model response of reduced real-world challenge would be transformed to an acceptable real-world response of the original challenge.

This created a nice theory from several points of view (see Chapter 3). Nevertheless, reality sometimes needs an adaptation of our mathematically nice formal model. We illustrate this using several practical situations, mainly in the area of recommender systems.

The data side of model situations is motivated by Fagin-Lotem-Naor's data model with attribute preferences and multicriterial aggregation. I thankfully acknowledge the influence of the late Peter Hajek [{H}]. Flexibility is obtained by offering top-k responses. We review our previous work on the preferential interpretation of fuzzy sets, implicit behavior in/and online/offline evaluation of recommender systems. I started to study these real-world situations during my involvement in the project [{NAZOU}]. We finish the paper with smart extensions of industrial processes. We propose a synthesis of these and formulate some open problems.

The main contributions of this paper are reformulations of previous results to an adapted ChRF setting in

- data mining and identification of user model from a parametric family of models,
- interpreting implicit behavior of a user,
- prediction of online behavior based on offline data,
- object detection when there are no human-annotated data.

We formulate several problems both in the formal and applied settings.

2 Search reduced to a data model with preference aggregation

Our main motivation is content-based recommendation (typically in real on an e-shop). Recommendation means to offer a user (customer) an ordered list of objects computed in the model situation. Depending on the display, these can be top-10 (or top-k in general) in some preference ordering of objects for each user separately.

2.1 eFLN – extended Fagin-Lotem-Naor approach

User object preference usually depends on preferred values of attributes (properties). In what follows, we describe some special cases of preference representation. Some of them will be discussed in further chapters in connection with experiments in real-world situations. The object model is represented by a relational scheme $R(oid, A_1, ..., A_m)$, where A_i's are attributes with domains D_i. Set of objects is a subset of the Cartesian product of domains $O \subseteq \Pi D_i$.

In [{FLN}] R. Fagin, A. Lotem, and M. Naor describe a system FLN where each object has assigned *m*-many attribute score $x^o_i \in [0, 1]$. A Pareto order-preserving (see [{GMMP agg}]) aggregation (combination) function $t{:}[0, 1]^m \rightarrow [0, 1]$ assigns each object *o* an overall score $r(o) = t(x^o_1, ..., x^o_i, ..., x^o_m)$. So the ordering of objects is represented by natural ordering of overall score in a unit interval of natural numbers as an aggregation of attribute score (ordering on attribute values). The main motivation of [{FLN}] was to describe a middleware system, where attribute score is available from a web service represented by a list of objects ordered descending by score either by an ordered approach or (when the ID of an object is known) by a random (direct) access. [{FLN}] presents a top-k algorithm and proves beautiful optimality in the price of sequential and random access over any possible algorithm, correctly finding top-k without random guessing.

As our interest is a content-based recommendation, we extend this approach by describing how these scores can be obtained. Assume, for each user $u \in U$ we have an attribute preference function $f_i^u{:}D_i \rightarrow [0, 1]$ and an aggregation function t^u.

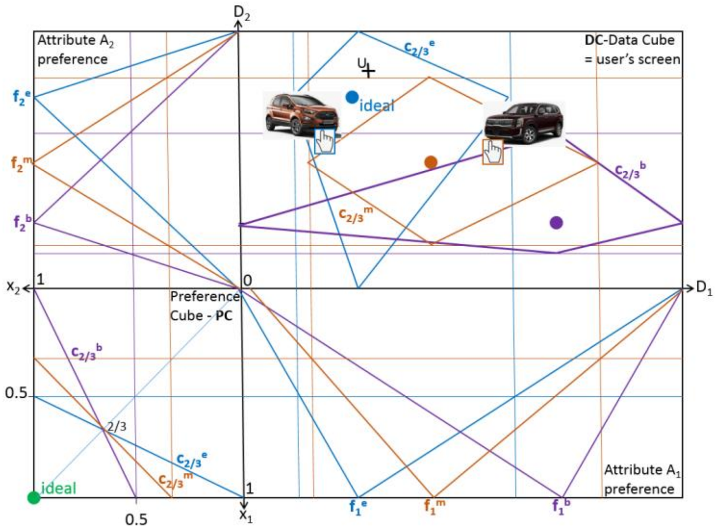

Fig. 1. Dynamical aspects of 3 sessions (**b**eginning, **m**iddle, and **e**nd) of a simplified linear two-dimensional eFLN model of preferences, see also [{KV ComSIS}]. Note, this can be used both inductively and deductively, in both directions from PC to DC and DC to PC.

The overall preference $r^u(o)$ of an object *o* is given by

$$r^u\,(o) = t^u(f_1{}^u(oid.A_1),..., f_i^u(oid.A_i),... f_m{}^u(oid.A_m)) \quad (1)$$

Our system is an extension of the [{FLN}] approach as setting $f_i^u(oid.A_i) = x^o_i$ gives the original FLN system.

Illustration in Figure 1 can serve as a mock-up of an idea where data cube-DC (NE quadrant) is the user's screen (reality), graphically calculated from preference cube-PC (via SW, NW, and SE) 2/3 contour lines (also motivated by [{R}] and [{Br}]).

User's action (orange and blue click) changed the preference model and showed unseen objects in the estimated highly preferred area computed by a recommender. Using a geographic intuition, we depict 2/3 contour lines of *t* in preference cube-PC (see SW-south-west quadrant of Fig.1.); these can be translated to areas in data cube-DC (see NE-north-east quadrant) corresponding to objects with preference at least *2/3*. For a fixed user, dynamical illustration starts with $t^b(x_1, x_2) = (x_1+ 2x_2)/3$ begin (purple) session, via medium session (t^m is average in orange) to end one (blue) with $t^e(x_1, x_2) = (2x_1+ x_2)/3$. Note that NE quadrant with DC represents a real-world situation (on the user's screen), and the remaining quadrants SE with f_1, NW with f_2, and SW with aggregation represent the model situation. In the following chapter, we introduce a general framework for reductions of real-world situations to model situations. Further, several inductive aspects of eFLN will be dealt with in future chapters on real-world data.

2.2 Learning by identifying user's preference model

In the previous paragraph, the overall object preference score was a number obtained as an aggregation of an object's attribute preference score in a deductive model. An interesting problem is the learning of user preference model. We usually first learn attribute preference functions $f_i^{u,}$ and having these, we can estimate t^u. Another point is what we know about a user. This will be our task for the rest of the paper.

Here we mention results from an unpublished preprint [{PEV-TFS}], partly published in [{PEV 2015}]. We had real-world production data with individual purchases. For learning f_i^u's we implemented several regressions and geometric heuristics. For learning t^u's we used identification of parameters of fuzzy t-conorms (S-norms, see [{GMMP agg}]). The final model was an aggregation of individual content-based models and an additional aggregation of behavioral data overall users. See Table 1. where the best results evaluated by nDCG and position resp. metric are depicted.

Table 1. Results of best methods of aggregation identification and regression [{PEV-TFS}]

Content-based individual		Behavioral all users		
Aggregation	attribute	aggregation	attribute	metrics
Frank	linear	Sugeno-Weber	linear	nDCG
Schweizer-Sklar	quadratic	Sugeno-Weber	linear	position

That is, when the overall efficiency of the system is evaluated by nDCG metric, the best results were attained by tuning parameters of the Sugeno-Weber family of conorms over two inputs:

1. tuned parameters of Frank conorms over partly linear (triangle) estimation of attribute preferences of content-based individual preference models optimized according to the prediction of purchases with
2. aggregating the former with an estimation of attribute preferences of all users.

Similar results were obtained when the overall quality was measured by the position of the best object in testing data compared to its position in the prediction. It may be interesting to compare these early results with that of later publications ([{PV FUZZ, Jo}]).

3 ChRF – Challenge-Response Framework

The origin of the Challenge-Response Framework was an old mathematical idea of Galois-Tukey connection of [{V-GT}][2] in set and category theory. A. Blass in [{B-QA}] interpreted this as complexity reductions in theoretical computer science (later he calls it challenge-response reductions, see [{B-HST}] and Figure 2(a)). First, we define it and discuss it formally. Later we develop it in different real-world situations.

3.1 A formal model of Challenge-Response Framework

A Challenge-Response Situation $S = (C, R, A)$ consists of a set of challenge instances C, a set of possible responses R and a (possibly graded) binary acceptability relation $A \subseteq C \times R$. For a challenge instance $c \in C$ and a response instance $r \in R$ we read $A(c, r)$ as "r is an acceptable response to challenge c" (or also another reading "response r meets challenge c"). Please note that acceptability relation can be a function (algorithm, process ...).

Challenge-Response Reduction of a situation $S_1 = (C_1, R_1, A_1)$ to a situation $S_2 = (C_2, R_2, A_2)$ consists of a pair of functions (φ^-, φ^+) such that $\varphi^-: C_1 \rightarrow C_2$ is a reduction of S_1 challenges to S_2 challenges and $\varphi^+: R_2 \rightarrow R_1$ is a reduction of S_2 responses to S_1 responses. A quite natural requirement of equation (2) says that an S_2 acceptable response r to $\varphi^-(c)$ is reduced to an S_1 acceptable response to the original challenge, in a logical formula

$$(\forall c \in C_1)\ (\forall r \in R_2)\ (A_2(\ \varphi^-(c), r) \Rightarrow A_1(c, \varphi^+(r))\ ) \qquad (2)$$

In case that $A_2 = \alpha$ is an algorithm, the equation (2) changes to the following requirement

$$(\forall c \in C_1)(\ A_1(c, \varphi^+(\alpha(\varphi^-(c)))\ ) \qquad (3)$$

[2] Supported in 1990-91 by Alexander von Humboldt Foundation, Germany

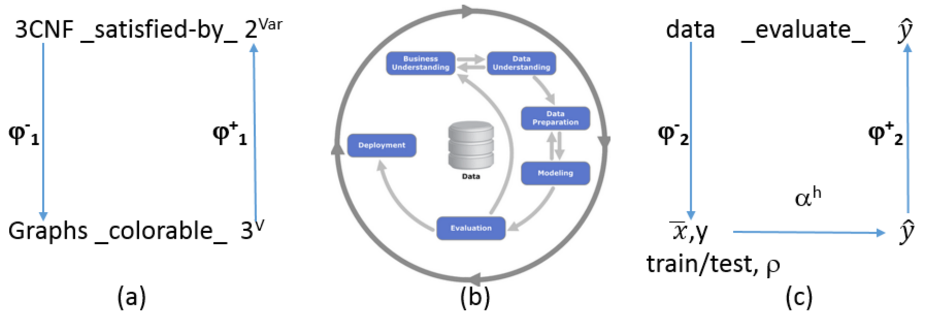

Fig. 2. ChRF illustration. Left: a ChRF reduction of 3SAT to 3COLOR. The middle (b) shows the process diagram for data mining[3]. Right (c) shows supervised learning reformulated in the language of ChRF language

Several aspects of this framework will be discussed.

First is the truth value of the formula in equation (2). In a pure logical understanding, it is easy to make it true. Just send all 3CNF instances to a graph that is not 3 vertex colorable; hence $A_2(\varphi^-(c), r)$ will be false, and the whole implication will be true (false implies * is always true). A. Blass in [{B-QA}] assumes that acceptability relations have domains *dom(A) = C*. In this case, there are no uncolorable graphs. In [{KV ComSIS}] and [{VV-CLMPST}]) we discuss the possibility to extend each response set with an extra element "*nar* = no acceptable response" and extend the acceptability relations by *A(c, nar)* for each $c \in C \setminus dom(A)$. It is shown that equation (2) with this nar-extended situation fulfills complexity reduction requirements.

We briefly mention another possible view. Consider S_1 situation as above as a (real world) challenge instance and S_2 as a (model) response instance of a generalized situation $\mathcal{S}$. Response S_2 meets challenge S_1 (is a good model for a real-world challenge) when there is a ChRF reduction (φ^-, φ^+) fulfilling equation (2). We can say that a generalized situation $\mathcal{S}_1 = (S_{11}, S_{12}, (\varphi^-_1, \varphi^+_1))$ is reduced to a generalized situation $\mathcal{S}_2 = (S_{21}, S_{22}, (\varphi^-_2, \varphi^+_2))$ (saying that he model $\mathcal{S}_2$ is better than model $\mathcal{S}_1$) when there is a pair of mappings (ψ^-, ψ^+) enabling to transform solutions of $\mathcal{S}_2$ to solutions of $\mathcal{S}_1$. In such a way, we can build a framework of *generalized Challenge-Response reductions gChRF*. This can be interesting both from a formal point of view and in practical applications.

3.2 ChRF as reduction of real-world situations to model situations

We would like to use ChRF idea in practical situations. The main viewpoint is that in a situation when one needs help, the recommendation we can reduce this to a model. So rephrasing, a Challenge-Response Reduction of a situation $S = (C_{real}, R_{real}, A_{real})$ to a model situation $S = (C_{model}, R_{model}, A_{model})$ consists of a pair of functions (φ^-, φ^+) such that $\varphi^-: C_{real} \rightarrow C_{model}$ and $\varphi^+: R_{model} \rightarrow R_{real}$ with a requirement that acceptable model

3 https://en.wikipedia.org/wiki/Cross-industry_standard_process_for_data_mining

responses are acceptable to original challenge (after reductions *(φ^-, φ^+)*). If A_{model} is an algorithm, α^h respective code can look like:

```
FOR each C^real from challenges
   CALL φ- with C^real RETURNING C^model
   CALL α^h with C^model RETURNING R^model
   CALL φ+ with R^model RETURNING R^real
   CALL A_real with C^real and R^real RETURNING accepted
   IF accepted PRINT "R^real is a response to C^real"
      ELSE PRINT "there is no response to C^real"
   END IF
END FOR
```

We have already seen that a pure mathematical understanding of equation (2) has to be adapted.

The second problem is how to understand quantifiers *($\forall c \in C_1$) ($\forall r \in R_2$)*. We will see that these can be interpreted as aggregation in the sense of various metrics used in experiments.

Inductive ChRF

Looking at Figure 2, we can see some similarities between CRISP-DM model and ChRF approach. Starting with the real situation first reduction can be to "business understanding." This can contain a challenge requiring reduction to "data understanding" and further to "data preparation." In [{KV ComSIS}] we introduced *Inductive ChRF* in which we look for a method $\alpha \in \Pi$ and a hyperparameter $h \in H^\alpha$ to evaluate α^h on training data $\bar{x}y$ comparing with $\bar{x}\hat{y}$. Here $\bar{x}y$ is an abbreviation of $c = \bar{x}, E(c) =$y where E is the example set and $,\hat{y} = \varphi^+(\alpha^h(\varphi^-(c)))$. The acceptability relation *evaluate($\bar{x}$,$\hat{y}$)* can be defined by an instance metric e.g., $|y-\hat{y}|$ and the quantifier ($\forall c \in C_1$) can be understood as an aggregation, e.g. by RMSE. The quality of our estimation is

$$\| (\forall c \in C_1)(A_1(c, \varphi^+(\alpha^h(\varphi^-(c))))) \| = \sqrt{\sum \frac{\left(E(c)-\varphi^+\left(\alpha^h(\varphi^-(c))\right)\right)^2}{|C_1|}} \quad (4)$$

The most usual case of finding an acceptable solution in a model situation is to find it by induction (data mining, learning ...). We are not going into details of ChRF modeling of learning, tuning, cross-validation etc.

Real-world acceptability depends on user u. In the case of recommender systems, this can be either user's explicit rating or our interpretation of u's behavior (see Chapter 4). User's behavior can be, e.g., purchase, click, time reading the detail of an item, etc.

Note that this gives a dynamic model of ChR, because user's satisfaction has to be followed (e.g., by scripts), evaluated, and taken into account in the next recommendation (more on this in chapter 5).

4 Implicit preference relations in recommendation

In this chapter, we use results from [{PV JoDS}] to make a step in extending ChRF. In previous chapters, the overall object preference was a number obtained as an aggregation of the object's attribute preference score. The size of a number itself does not matter. We use numbers as an ordinal scale and numbers code an ordering. For application in recommender systems, we sometimes need to aggregate several recommenders (algorithms). Sometimes these do not offer a rating (score). They give just a position (rank). Here we describe a real-world experiment where a recommender's linear ordering form was enhanced by a partial ordering coming from preference interpretation of the user's implicit behavior.

The approach of [{PV JoDS}] is illustrated in Figure 3 left. Fix a user u. Assume we have an ordered list of object $\overline{L_u}$, from a recommender. The idea is to use the information on the visibility of objects and the user's action (clicked, scrolled). In time T1 objects O1, O2, O3, and O4 were visible. He/she clicked on object O3 and did not act on the remaining object. This can be interpreted in a way that object O3 is more preferred than the other 3 objects. Nevertheless, after a scrolling (and much shorter visibility), objects O3 to O6 were visible, and there was no further action. Now object O4 was visible much longer than, e.g., O1. So some preference degree of O3 over O4 should be greater than that over O1. In [{PV JoDS}], we designed some measures to express this intensity and output relation IPR[4]. As user behavior data are quite sparse, we extended this relation by the similarity of the object and computed relation $\widehat{R_u}$ (it is a partial order, nevertheless has information on objects the user was searching). In [{PV JoDS}] we have designed several ways how to merge these two orders, the linear ordering $\overline{L_u}$ and the partial ordering $\widehat{R_u}$, to get final ordering L_u. Then in experiments, we have evaluated how far is L_u better than $\overline{L_u}$, and which method gives the best results. Roughly speaking, when, e.g., a contradiction between the ordering of O3 and O1 in L_u and $\widehat{R_u}$, is discovered we can put O3 just before O1 in the next iteration of L_u (or O1 just behind O3, or swap both ...). Please consult the paper for more details.

Here we are interested in an extension of the ChRF where the model (algorithm α_3) giving L_u is a combination of algorithms α_1 and α_2, originally computing $\widehat{R_u}$ and $\overline{L_u}$.

To our surprise, original meet and join in the algebraic category of {V-GT} and {B-QA} (or corresponding lattice) do not apply. In Figure 3 right, we propose a construction that takes responses of two models and presents them as a challenge of a model situation that could be considered an aggregation of previous situations. It is an interesting problem if this construction has a category-theoretic interpretation.

[4] IPR source codes: https://github.com/lpeska/Implicit-Preference-Relations , for more resources see the paper

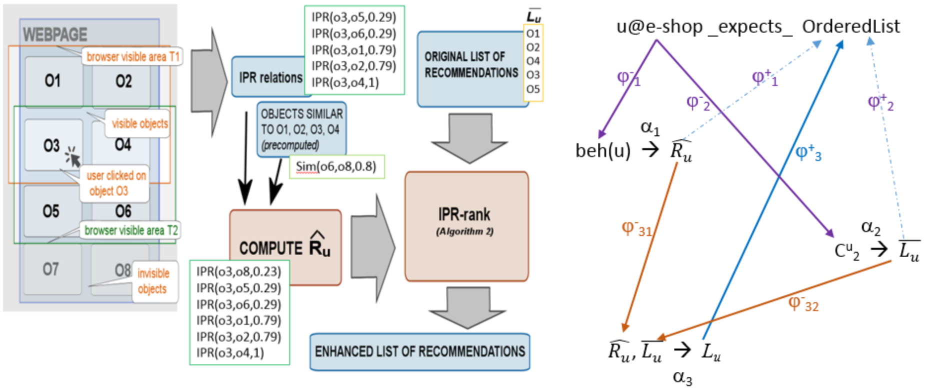

Fig. 3. Left, enhancing recommender results by implicit user's behavior (see [{PV JoDS}]). Right, a description using extended ChRF for this sort of aggregation.

5 Predictability of online recommendation

This is the last chapter devoted to recommender systems. We would like to illustrate here another possibility of interpreting universal quantification. The content is based on paper {PV FUZZ}[5]. The long-standing problem is the connection between offline recommendations (one based on historical production data) and online recommendations. Of course, we can be careful and use only A/B testing for changing our recommender. Still, each A/B test takes time, effort and can be discouraging for customers. So the idea is to provide A/B testing only with the promising candidate(s). Before choosing this candidate, we have to solve the problem of algorithms and metrics by which we will evaluate which candidate solution is most promising. And this was the main goal for {PV FUZZ}.

We had true production data and also the access to provide online A/B testing. Therefore, offline data played the role of a model, and online production was the real world to be modeled. The implication $A_{model}(\ldots) \rightarrow A_{real}(\ldots)$ became $A_{off\text{-}line}(\ldots) \rightarrow A_{on\text{-}line}(\ldots)$, and this can be interpreted as our main task – how to evaluate online results based on offline achievements.

The first problem occurs with users. It is difficult to identify users from offline data and online testing (these can be quite disjoint sets). So, we have to quantify all users. Quantification over all objects is already a part of ChRF reduction formalization. For the beginning, we chose several item-to-item recommendation algorithms sufficiently rich to represent a content-based attribute, textual description of objects, and collaborative aspects of our data. So finally, we had to quantify over all algorithms.

5 See https://github.com/lpeska/FUZZ-IEEE2020 for source codes, evaluation data and complete results are available from

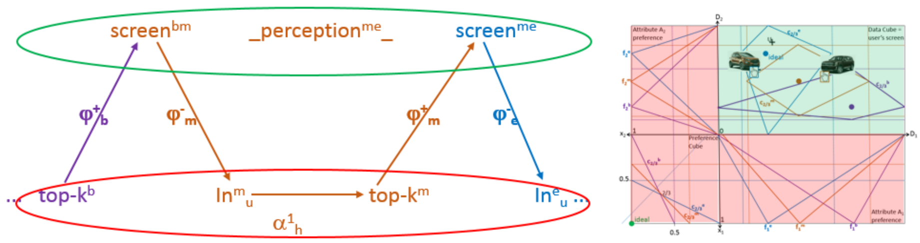

Fig. 4. Left, illustration of ChRF representation of using offline data for online prediction. Right there is miniaturized Figure 1 with 3 sessions, and the green background represents the real situation, whereas the red one the model.

In Figure 4, using the notation of beginning, middle, and end session, we can see that the previous session's responses (visualized top-k recommendation) are challenges of the next session. User's acceptability is his/her perception of sessions, φ^{-} denotes our scripts recording user's behavior. The model computes the next recommendation based on previous behavior and outputs top-k, which is visualized by φ^{+} to next session.

This is another understanding of ChRF in a real-world situation. In {PV FUZZ} we provided A/B testing with the 12 most promising algorithms. Motivated by {HH} we interpret quantifiers in implication describing ChR reduction from offline to online by aggregation. It makes good sense because we would like to have an overall evaluation of" how good are algorithms (trained offline with respect to some metric) in predicting user's online behavior." Most of the aggregations were just averages. It is a challenge for future research and experiments to consider some other aggregations. In {PV FUZZ} we aggregated over all algorithms, and results were metrics that are the best predictors for an online recommendation. One can imagine aggregating over metrics to get the best algorithms, and joining both could be interesting to test.

6 Object detection from visual data

So far, we have had more or less reliable data about real-world situations either from training and/or behavioral data. Data reliability is obtained by human expert intervention designing data collection. In the final chapter, we consider a situation where we do not have any human-annotated training data. We build on the object detection model developed in {BHV-ISM}. The main goal of {BHV-ISM} was to automatically create a system for object detection in industrial premises without any human intervention. This has to lead us to a concept of "pseudo ground truth." Pseudo ground truth PGT_3 is created by a heuristic process considering a correct object detection be the one where at least three models agreed (lower index 3 in PGT_3 refers to a number of models required to agree on an instance).

From the point of view of ChRF, this situation is interesting. The real ChR situation is on the camera screen. For the model situation, we do not have any train and test data (correct in the sense that the object detection bounding box and the class were annotated by a human). So, we have a pseudo-model situation, and the main point is that the modeling algorithm is chosen without any human intervention, just considering the performance using pseudo-ground truth. So it can be deployed in a situation where there is no staff for annotation. We discuss the quality of our model to give an acceptable response. For this purpose, we annotated some video frames.

Figure Fig. 5 shows an example of a CCTV camera from an office environment. The picture shows the detections of 11 objects belonging to 3 classes - person, bottle, and mouse. False-positive predictions are marked in red.

Fig. 5. Office scene with true and false positive object detection. Ids of persons will be used in the text below (source YouTube).

Figure 6 shows the performance of nine models we used for the creation of the pseudo-ground truth.

Models are in columns, and predictions are rows (ids of detected persons correspond to those in Figure 5). The last three rows are false positives.

The first 9 columns (with names of deep neural network models) depict the size of the confidence score of respective predictions in the blue bar. We can see that some models did not detect an object at all, some detected with small confidence, and some made a wrong prediction. GT column is a yes-no column (depicted in black) that shows objects made to pseudo-ground truth on this image. PGT_3 column depicts confidence the pseudo ground truth was obtained. Based on PGT_3, we chose the best model. The best model (YOLO3) for a specific CCTV camera (this model also shows a false negative error rate). In addition to the best model, the method also determines the order of the models according to the expected performance. Column W * TOP3 shows the confidence we gained by weighing the 3 best models obtained by our method. The weights for these 3 models were determined using linear regression. However, the CenterNet-HG104 model was also included in the TOP3 models, which demonstrates a false positive prediction in the case of the object I. This false-positive prediction was also transferred to the W * ALL prediction. To complete, we also present the W * ALL column, which shows the confidence gained by weighing all 9 models. Again, these weights are determined by linear regression. The hope was that the wrong influence of some models might have been eliminated. This elimination really came about because it is visible that false-positive predictions (J and K) from the Retinanet-RN50 model are not included in W * ALL.

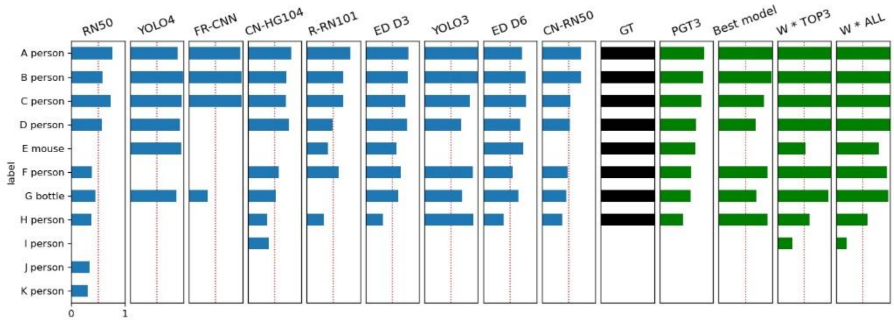

Fig 6. Different confidence of models on respective (true/false) detections.

This experiment shows a new ChRF reduction from a real situation to model one. Various metrics can evaluate the overall quality of responses. Some show that false positives can be eliminated. Still, there is a problem with false negatives – we leave it for future work.

7 Conclusions and future work

We considered real-world recommender systems based on the Fagin-Lotem-Naor data model with aggregation. Our primary interest was to test Challenge-Response Framework in several real-world situations. Results show that proper reduction to a model situation requires using different metrics and different aggregations in place of logical quantifiers and calculation of truth values. Finally, we mention several problems both in the formal model and practical use of ChRF.

Results show that ChRF is quite flexible when measured by appropriate metrics. The role of aggregation in quantification goes behind the classical understanding of aggregation in multicriterial modeling.

References

1. {NAZOU} Bielikova, M., Hluchy, L., Navrat, P., Pazman, R., Vojtas, P. A framework for acquisition, organization and maintenance of knowledge in an environment of heterogeneous information resources. http://nazou.fiit.stuba.sk, last accessed 2021/06/09
2. {B-QA} Blass, A. Questions and Answers. A Category Arising in Linear Logic, Complexity Theory, and Set Theory, in Girard, J.-Y. et al (eds.), London Mathematical Society Lecture Note Series 22, Cambridge University Press, Cambridge 1995, 61-81

3. {B-HST}Blass, A. Combinatorial Cardinal Characteristics of the Continuum. In: Foreman, M., Kanamori, A. (eds.) Handbook of Set Theory, pp. 395-489. Springer, (2010)
4. {BHV-ISM} Brezani, S., Hrasko, R., Vojtas, P. Smart extensions to regular cameras in the industrial environment. Preprint, submitted to ISM'21
5. {Br} Brown, T. Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation. HarperBusiness (2009)
6. {FLN} Fagin, R., Lotem, A., Naor, M. Optimal aggregation algorithms for middleware. Journal of Computer and System Sciences, 66(41), 614–656 (2003)
7. {GMMP agg} Grabisch, M., Marichal, J., Mesiar, R., & Pap, E. Aggregation Functions (Encyclopedia of Mathematics and its Applications) Cambridge University Press. (2009)
8. {H} Hajek, P. Metamathematics of Fuzzy Logic. Springer (1998)
9. {HH}Hajek, P., Havranek, T. Mechanizing Hypothesis Formation. Springer (1978)
10. {KV ComSIS}Kopecky, M., Vojtas, P.: Visual E-Commerce Values Filtering Framework with Spatial Database metric. Comp. Sci. Information Systems 17(3), 983–1006 (2020)
11. {PEV-TFS}Peska, L., Eckhardt, A., Vojtas, P. Interpreting Implicit User Behavior for E-shop Recommendation with Families of Fuzzy T-conorms, 36 pages, preprint 2012
12. {PEV 2015} Peska, L., Eckhardt, A., Vojtas, P. Preferential Interpretation of Fuzzy Sets in Recommendation with Real E-shop Data Experiments. Arch. Philosophy and History of Soft Computing 2(2015) 14 pages, https://www.unipapress.it/it/book/aphsc-2-%7C-2015_170/
13. {PV FUZZ} Peska, L., Vojtas, P. Predictability of Off-line to On-line Recommender Measures via Scaled Fuzzy Implicators, In FUZZ-IEEE 2020, pp. 1-8,
14. {PV JoDS}Peska, L., Vojtas, P. Using Implicit Preference Relations to Improve Recommender Systems. J Data Semant 6, 15–30 (2017)
15. {R} Ries, E. The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses. Crown Publ. (2011)
16. {V-GT} Vojtas, P. Generalized Galois-Tukey connections between objects of real analysis, Israel Mathematical Conference Proceedings 6, 619-643 (1993)
17. {VV-CLMPST}Vojtas, M., Vojtas, P. Problem Reduction as a general epistemic reasoning method. In Extended abstract CLMPST 2019, EasyChair Preprint no. 1208, https://easychair.org/publications/preprint/HfsP , last accessed 10/06/2021