Predicting the Outcome of Ongoing Automated Negotiations

Tamara C.P. Florijn[1,2][0009−0004−2300−9291], Mick Tijdeman[2][0009−0001−8798−0376], Pınar Yolum[0000−0001−7848−1834][2], and Tim Baarslag[1,3][0000−0002−1662−3910]

[1] Centrum Wiskunde & Informatica
{tamara.florijn,t.baarslag}@cwi.nl
[2] Utrecht University, The Netherlands
p.yolum@uu.nl
mick.tijdeman@gmail.com
[3] Eindhoven University of Technology

Abstract Automated negotiation offers an efficient way to reach agreements between multiple parties. Estimating the outcome of a negotiation before it is finished allows a party to take effective actions, for example by ending negotiations when prospects for a favorable outcome are small. However, estimating the outcome is difficult as many (uncertain) factors affect the course of a negotiation. Accordingly, this paper presents a method for predicting the outcome of ongoing bilateral negotiations called *PrONeg*. From the perspective of one agent, we predict the future trajectories of its own and its opponent's bids using time series forecasting methods. These two forecasts are used to find the agent's utility outcome distribution, along with the probability of reaching an agreement by the end of the negotiation. Finally, we integrate the scenario with the agent's utility prediction for a probability prediction of specific scenario outcomes. Our experiments show that Gaussian processes perform best in most settings, including balancing predicting true breakoffs without misclassifying agreements as such. Overall results are encouraging and suggest that integrating opponent model information could further enhance prediction accuracy. A typical future use case could combine PrONeg with human negotiation forecast techniques to serve as a negotiation support system.

Keywords: Automated negotiation · Outcome prediction · Time series forecasting

1 Introduction

The field of *automated negotiation* researches efficient ways to reach acceptable agreements with multiple parties, applied in e.g. procurement [1,23], energy market [5] and supply chain management [25]. However, not all automated negotiations end in acceptable agreements. Often, parties invest a lot of time and energy into trying to reach an agreement, only to find themselves failing to find one in the end. Thus, estimating the outcome of a negotiation before it is finished is useful as it could signal an unlikely agreement to the agent and thus save them considerable time.

Consider the following example of a negotiating fruit buyer. A specific outcome (e.g., 10 pears for price 3) has a certain value for the negotiator called *utility*. The fruit buyer wants to know if the ongoing negotiation is worth the time, or whether they should quit. An early prediction of a non-satisfactory utility outcome or a failed negotiation can save them time and thus is resource efficient. The outcome of the negotiation influences decisions outside the current negotiation as well, so early outcome prediction facilitates proactive coordination of parallel actions. For example, the fruit buyer wants to know what they expect to spend, because they need to set a budget for other parallel negotiations.

However, predicting whether a negotiation ends in agreement, and, if so, predicting the agent's utility of the outcome and the specific outcome of a negotiation, is hard. Firstly, it is crucial to balance predicting breakoffs accurately without misclassifying agreements as failures, which can lead to lost opportunities. Secondly, even when an agreement is correctly anticipated, accurately determining the corresponding utility is still hard. An agreement could be the immediate next bid, or lay far into the future, but where that exact point is depends on the course of bidding and the (unknown) strategy of both agents. The courses of bids of both agents are interdependent, with each concession influencing the other party's subsequent moves. This creates a complex dynamic, especially since the preferences of the opponent are private. Furthermore, not only are the opponent's preferences unknown to the agent, but the opponent does not have access to information about the agent's preferences and strategy either. This creates an impression of randomness in the received utility of the opponent's bids, as they are trial and error. Finally, assuming an accurate prediction of the outcome utility, the specific outcome remains challenging to pinpoint, because different outcomes that are close to each other w.r.t. the agent's own utility can be far apart w.r.t. the opponent's utility. If the number of outcomes is large, it becomes difficult to identify the exact outcome.

Existing research has been dedicated to predicting the outcome in human-human negotiation [26,28] and hybrid human-agent negotiation [8,22]. These results cannot be applied in our automated negotiation setting, as their analysis is focused on factors specific for humans, such as emotional pointers and utterances, that are not present in automated negotiations. Previous research in automated negotiations focuses on predicting offers in advance, e.g., the expected counteroffer [7], but the outlook of one step into the future does not provide enough information about the final outcome. Some authors predict the opponent's concession curve [29] or research opponent negotiation strategies [6,9,19]. Our approach advances the goal of predicting noisy bidding curves to pinpointing the outcome of a negotiation. While Moosmayer et al. [26] retrospectively analyze important factors that predict the outcome, such as the level of reservation value, our model predicts the outcome during the negotiation. Although these correlations could be useful inputs for our model, they do not provide the tools to predict the outcome in an ongoing negotiation.

We propose a modular, online, risk aware prediction method called PrONeg for automated negotiation outcome prediction. Our approach is modular, in the sense that it can be applied by any type of agent in bilateral negotiation. This is achieved by only relying on the incoming bids of the current negotiation, without requiring that the agent has an opponent model or any other necessary training phase in advance. During the negotiation, our outcome prediction method can be used as a tool for agent designers to guide their strategy, using only the information available at that point, saving the negotiator time and energy at an early stage. Risk estimation can aid the agent to make the trade-off between capturing as many breakoffs as possible and losing opportunities by ending potentially fruitful negotiations.

Our method PrONeg is a pipeline consisting of three parts, as visualized in Figure 3. Firstly, from the perspective of one agent, the opponent's utility curve is predicted using time series forecasts. We regard the history of received bids as a time series of utility values from the perspective of one agent. We also predict the agent's own utility curve. Secondly, we intersect these two predictions using Monte Carlo sampling to find a predicted distribution of outcome utilities. We sample from both distributions and estimate the likeliest points of agreement. We convert the sampled points into a density distribution of outcome utilities and an agreement probability. Thirdly, we combine the predicted outcome utility distribution from the previous step with the specific scenario to determine how likely each specific outcome in the scenario is. We evaluate our method in a rich setup that considers negotiations between different types of agents with different characteristics. We find that predictions of negotiations with agents that use opponent models are more accurate, and that predictions made closer to the actual agreement time are also more reliable. The specific outcome prediction results are promising and can be further enhanced by incorporating opponent model information.

The overview of the paper is as follows. Section 2 introduces and formalizes the notion of negotiation and concession to define the problem of outcome prediction. We present the layout and formalization of our proposed method PrONeg for outcome prediction in Section 3, followed by an experimental evaluation of PrONeg applied to a data set of negotiations (Section 4), and a discussion looking ahead to future research opportunities (Section 5).

2 Problem Setting

We consider a setting where one agent aims to predict the outcome in a bilateral negotiation. An often used protocol for bilateral negotiations is the Alternating Offers Protocol (AOP) [27]. According to this protocol, two agents take turns in making bids, until one of the agents accepts the bid, ends the negotiation early or until deadline D is reached. If no agreement is reached before deadline D, the negotiation ends in *breakoff*.

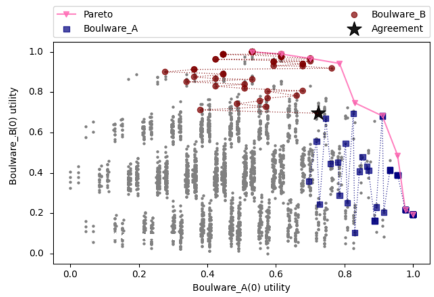

Figure 1: The bidding curves of two agents in NegMAS, where the lines with squares and circles show the utilities of the bids made by the agents.

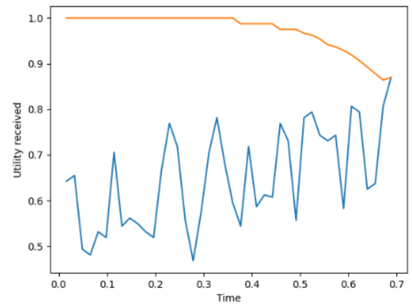

Figure 2: Bidding curves from the perspective of a single agent: in orange, the agent's own bids (upper line), in blue the opponent's bids (lower line).

Formally, we can formulate a negotiation with the AOP protocol as the sequence of exchanged bids:

$$\mathbf{b} = (b_1^1, b_1^2, b_2^1, b_2^2, \dots, b_{r'}^x, b_r^y),$$

with b_i^a the bid made in round i by agent a. The last bid in round $r \leq D$ can be posed by either Agent 1 or 2, meaning that $x, y \in \{1, 2\}$ and $x \neq y$. If the last agent replies with the same bid, so $b_{r'}^x = b_r^y$ with $r' = r - 1$ or $r' = r$, then the negotiation ends in agreement and has b_r^y as outcome. Offering the empty set $\varnothing$ is interpreted as ending the negotiation early, associated with a bid sequence where $b_r^y = \varnothing$. Both agents have preferences over what the outcome of a negotiation is, modeled using a utility function. Each bid b in outcome space Ω has an associated utility for both agents, which is a value between 0 and 1 calculated using an additive utility function $U_a : \Omega \to [0, 1]$ for agent $a \in \{1, 2\}$.

An agent seeks the outcome that best aligns with its preferences and maximizes its utility. However, since both agents have different preferences and must agree on the outcome, agents cannot easily get the result they most desire. Instead, they attempt to find an outcome that satisfies both agents through concession. By sacrificing some of their own utility, they try to align with and appeal to the other agent's interests. Both agents may start at their best option (maximum utility) and then slowly explore other options while conceding small parts of their utility until an agreement is reached. From an outside view with full knowledge of both utility functions, these concessions may look like Figure 1. As the agents do not know the preferences of their opponent, the utility of the bids show a chaotic, 'trial-and-error' curve in terms of the opponent's utility. The outcome of the negotiation is when the two agents concede enough to appeal to the others wishes and 'meet in the middle': the point of agreement.

What that point of agreement will be, could be valuable information for the agent during the negotiation. It is hard to find good outcomes, and it often takes a lot of time to find mutually acceptable outcomes, since utility functions are private, the opponent's strategies are unknown, and the outcome space can be large. If the negotiation has little chance of a good ending, it could be beneficial to end the negotiation and save resources. In addition to predicting the chance of success, it can also be valuable to predict the specific outcome of a negotiation, as that can affect what is optimal in decisions outside the negotiation. For example, a monthly budget imposes constraints on multiple purchases. The agent could already act on the expected outcome of the current negotiation and align different actions well to find good outcomes.

The information we have to estimate the outcome is limited, especially because we assume no prior knowledge of the opponent's preferences. The data that is accessible include the agent's own bids and those received from the opponent, along with their associated utilities from the *perspective of the agent*, while the opponent's utility of the bid history remains unknown. An example visualization of what is available can be seen in Figure 2, showing the agent's own bidding curve (upper line) and the opponent's bidding curve from the agent's perspective (lower line). In this example, the agent has a clear downward concession curve, while the opponent is seemingly trying a wider palette of offers. The opponent starts low, as their preferences usually contrast. Since the opponent has no information on the preferences of the agent, it starts by trial-and-error, resulting in a seemingly chaotic bidding pattern. As time increases, the trend of the opponent's bidding curve goes up. This is due to their gradual willingness to concede and their ability to learn about the agent's preferences over time. As time continues further, the two bidding curves may meet in a point of agreement.

Predicting the utility of point of agreement is a challenging endeavor because of the noise of incoming bids and uncertainty about the opponent's preferences and strategy. Even if the prediction of the utility of a negotiation is accurate, the specific outcome is still hard to find, because different outcomes that are close to each other in utility for one agent, can be far apart for the other. In a large outcome space, it is even more difficult to find the exact one outcome. All this makes predicting negotiation outcomes non-trivial.

In essence, the task of predicting the negotiation outcome is finding the point where the utility received from the opponent and the agent's own concession strategy intersect. We differentiate between three aspects of the outcome of the negotiation: (1) the probability of reaching any agreement at all, (2) if there is an agreement, the utility of the outcome in expectation, i.e., the expected utility, (3) the probability of the specific outcome itself. These aspects are predicted given the history of bids in an ongoing negotiation at round $k \leq D$ from the perspective of Agent 1.

3 PrONeg: Outcome Prediction Method

We propose a pipeline for outcome prediction called PrONeg (**Pr**edicting the **O**utcome of a **Neg**otiation) consisting of three steps, which is visualized in Figure 3. Firstly, we need to predict the course of negotiation, predicting both the utility of the bids of the opponent and the utility of the agent's own bids using time

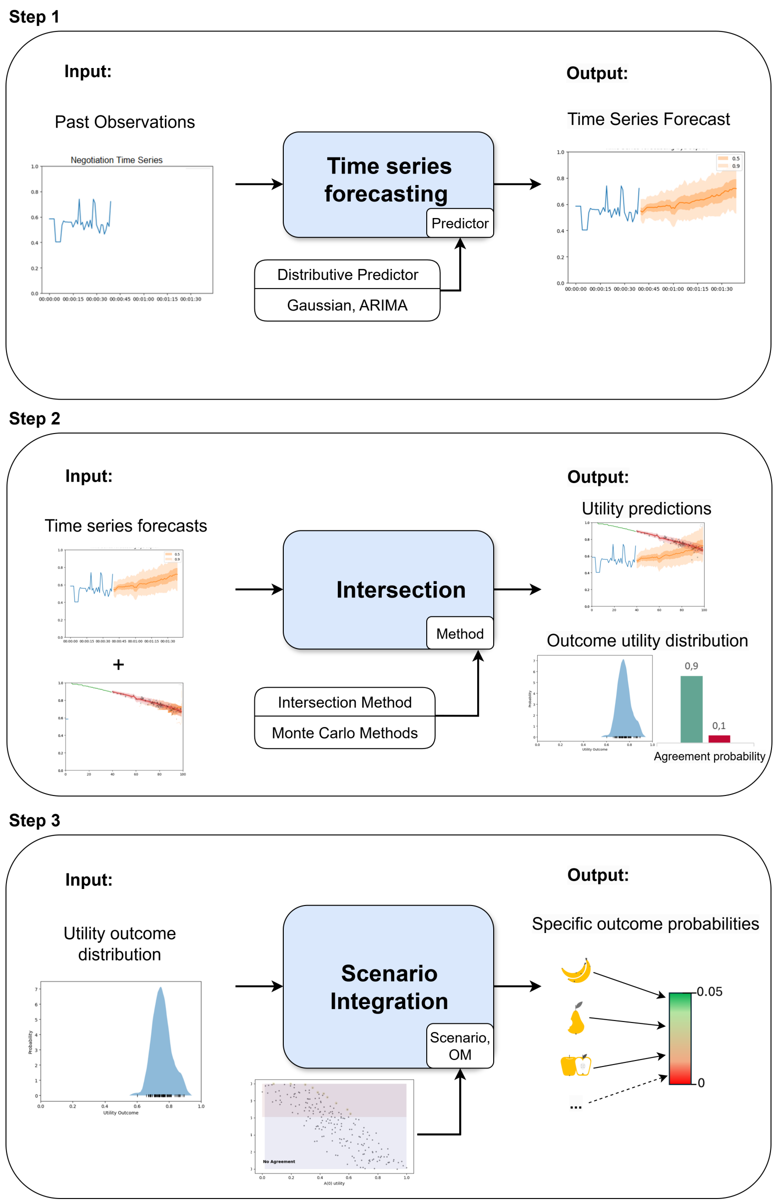

Figure 3: Overview of the proposed pipeline PrONeg in three steps.

series forecasting (Step 1 in Figure 3, Section 3.1). Secondly, we need to construct a distribution over the utilities of the potential outcomes using an intersection of the two predictions (Step 2 in Figure 3, Section 3.2). Thirdly, based on the outcome utility distribution, we aim to find the predicted probability over all specific outcomes in the scenario (Step 3 in Figure 3, Section 3.3).

3.1 Step 1: Time Series Forecasting

Bids posed by the agents can be modeled as a series of data points of utilities ordered in time, as in [29,31]. Formally, we write two utility sequences from the perspective of Agent 1, u_1^1 with utilities of bids posed by itself, and u_2^1 with utilities of bids posed by the opponent, Agent 2, as follows:

$$\mathbf{u}_1^1 = \big(U_1(b_1^1), U_1(b_2^1), U_1(b_3^1), \ldots, U_1(b_r^1)\big),$$

$$\mathbf{u}_1^2 = \big(U_1(b_1^2), U_1(b_2^2), U_1(b_3^2), \ldots, U_1(b_r^2)\big).$$

When we model negotiations as numbers ordered in time, they become time series, where every round is seen as one unit of time. Therefore, we can apply methods from the field of Time Series Forecasting (TSF) to predict their future values. TSF is a broad field within statistical analysis that contains a variety of techniques. Classical TSF methods rely on the careful tuning of a model, taking into account statistical parameters like trend and seasonality to design the perfect model for the time series at hand.

Formalization. The time series over a specific interval $\mathbf{u}^{\mathbf{a}}_{\mathbf{s:r}}$, where a is the associated agent, s is the start of the interval and r the end, is defined by

$$\mathbf{u}^{\mathbf{a}}_{\mathbf{s:r}} = \big(U_a(b_s^a), U_a(b_{s+1}^a), \ldots, U_a(b_r^a)\big).$$

Given a time series observed at time t with a maximum length of r, we find a function f to estimate a distributional forecast $\hat{\theta}$ for every time step in the interval $[t : r]$:

$$f(\mathbf{u}^a_{1:t}) = (\hat{\theta}^a_{t+1}, \hat{\theta}^a_{t+2}, \ldots, \hat{\theta}^a_r) = \boldsymbol{\hat{\theta}^a},$$

where $f : S^* \longrightarrow S^{**}$, with S^* a set of (utility) values for which each value s in S^* holds $s \in [0, 1]$, and S^{**} is a secondary set where each value is a distribution θ. We note that different regression or forecast techniques can be used to find this function f, provided that their output is a distribution. Even though our pipeline is tailored to non-learning distributive methods, trainable algorithms like Deep Neural Networks can be integrated easily [31], as well as point-wise predictors such as linear regression and exponential smoothing, that form peak distributions aggregating to a line over all time steps.

3.2 Step 2: Intersection

When both agents agree on an offer, they have conceded enough to accept the utility corresponding to that bid: they meet each other ‘in the middle’. In the context of bidding curves, we define an agreement between two agents as the point when the forecast bidding curves of the two agents intersect each other. This intersection indicates that both agents have conceded enough to reach a mutual agreement around this utility score.

We generate possible negotiation scenarios to determine expected points of agreement in terms of utility. Our utility forecasts produce a distribution over possible curves; we can think of these as different future scenarios, both over the agent's curve and the opponent's curve. Some combinations of these bidding curve scenarios end in failure, others end in agreement with varying utility outcomes. To reflect these bidding curve scenarios, we take random samples from the distribution using Monte Carlo sampling. We sample from the agent's and its opponent curve to construct a negotiation scenario. We inspect whether these two curves intersect, and if so, what this point of intersection would be. By repeating this sampling, we construct a set of possible outcome utilities. We then fit a probability density distribution to this set.

Not only does Monte Carlo provide insight in the outcome utility distribution, it also produces an agreement probability. By sampling numerous times from both bidding curves, we can keep track of the number of times an outcome is reached. We average over all sampled scenarios and translate it into an agreement probability forecast. We calculate the fraction of sampled combinations that end in agreement. If only part of the sampled combinations end in agreement, the agreement probability is strictly between 0 and 1, which allows the agent to do a risk estimation on the chance of success.

Formalization. We use multiple agreement predictions generated through Monte Carlo sampling to estimate the overall agreement probability p and construct an approximate utility probability distribution $\hat{\beta}$, which quantifies the likelihood of reaching an agreement and characterizes the expected outcome utility. A generated forecast scenario is labeled as an agreement if the predicted utility curve of the agent is lower than the predicted utility curve of the opponent before deadline r, that is $u_t^1 <= u_t^2$ given $u_{t-1}^1 > u_{t-1}^2$, with $u_t^a \in \mathbf{u}_{\mathbf{s:r}}^{\mathbf{a}}$.

Given two distribution vectors of the remaining rounds $\hat{\boldsymbol{\theta}}^{\mathbf{1}}$ and $\hat{\boldsymbol{\theta}}^{\mathbf{2}}$ for Agent 1 and 2, respectively, i.e., a predicted distribution of utility at each remaining time step for both agents, we predict the agreement probability p, and estimate a probability density distribution $\hat{\beta}$ over the outcome utility range $[0 : 1]$. The goal is to find an estimator function f such that

$$f(\hat{\boldsymbol{\theta}}^{\mathbf{1}}, \hat{\boldsymbol{\theta}}^{\mathbf{2}}) = (\hat{\beta}, p),$$

with $f : S^{**} \times S^{**} \longrightarrow (B, [0, 1])$, where B is the set of all possible continuous distributions in the interval $[0, 1]$. If both input distributions are peak distributions in the form of a line, observe that the agreement probability is either 0 or 1, given that two lines either intersect at one point or are parallel and never intersect.

3.3 Step 3: Scenario Integration

The predicted outcome utility can help the agent improve strategic decisions, for example to decide on ending the negotiation. To assist an agent to decide what strategies to pursue in parallel negotiations, merely the predicted utility is not enough. Therefore, we extend our prediction on outcome utility to predict the probability of specific outcomes, linking back to the scenario of the negotiation.

We propose to assign a probability to each outcome based on the probability density distribution of the predicted outcome utility of step 2. If an outcome utility α has a high associated value in the probability density distribution, then an outcome with utility α would intuitively also have a high chance of realization. However, an outcome with a high utility from the perspective of the agent may not necessarily be so for the opponent. The information of an opponent model would be useful to distinguish between these, if available, to achieve a higher accuracy.

Formalization. Let $\hat{U}_2$ be the opponent model, i.e., the estimated utility function of the opponent, Agent 2. For specific outcome prediction, we look for a function f based on the probability utility outcome distribution $\hat{\beta}$ and optional opponent model $\hat{U}_2$ such that for all outcomes ω in the outcome space Ω:

$$f_{\hat{\beta},\hat{U}_2}(\omega) = \hat{P}(\omega),$$

with $\hat{P}(\omega)$ the predicted chance that ω is the outcome of the negotiation. Note that the specific use of these probabilities depends on the goal of the agent designer. For instance, an agent aiming to estimate its expected utility needs to compare the relative likelihoods of outcomes, which requires using the probabilities directly. In contrast, if an agent's designer has a strategy to select a bid from the top 10% most likely outcomes, a ranking of all outcomes is sufficient. A straightforward way to create a ranking is to list all outcomes in decreasing order of density, an approach also used in the experiments of this paper.

4 Experimental Evaluation PrONeg

This experiment showcases an implementation of our outcome prediction pipeline PrONeg, and aims to evaluate the performance of different TSF methods over a large variety of settings. We evaluate the breakoff prediction and the estimated utility of the outcome of all TSF methods in combination with Monte Carlo sampling. Finally, we test the scenario integration step by evaluating the likeliness ranking of the real outcome.

4.1 Experimental Setup

We build a dataset of 8500 negotiations by running tournaments with different types of bilateral agents using the well-known negotiation simulation platform NegMAS [24].The first tournament is run between classic time-dependent agents [12] (denoted by *TDA*), which shows a broad variety of bidding curves, determined by

$$u = 1 - \big(t^e \cdot (1 - m)\big),$$

where t is the fraction of the total negotiation time that has passed, e is the concession exponent and m is the minimal acceptable utility. These time-dependent agents use a static opponent model, assuming that the opponent's utility is the opposite of their own. Furthermore, we introduce an agent type by extending the TDA with the opponent model from 2011 ANAC winner *HardHeaded* [17] and the opponent model of ANAC 2012 winner *CUHKagent* [15][4], denoted by *TDA-HH* and *TDA-CUHK*. To simulate more advanced opponent models, we also conduct tournaments with time-dependent agents that use a module offering partial yet accurate information about opponents, improving over time. It enables the agent to disregard unfortunate bids where the opponent's utility is below $p \cdot t$ for $p = 0.25$ and $p = 0.5$, referred to as *TDA (0.25)* and *TDA (0.5)*.

To effectively test the method's performance in predicting agreement probability, negotiations should result in both agreements and breakoffs: we target at a breakoff rate of 15% to 30%. Preliminary experiments show that agents with exponents between 0.5 and 8, and minimal acceptable utility between 0.5 and 0.7, meet this target.

Finally, we run a tournament with four winners of the Automated Negotiating Agents Competition (ANAC) 2011, as the bilateral negotiation setting of ANAC 2011 is the most similar to our setting, providing agents with a relative time indication based on the remaining rounds and using a time discount of 0. We include *HardHeaded*, *AgentK2*, *IAMhaggler2011* and *TheNegotiator* [14] using the Genius-bridge in NegMAS, referred to as *ANAC*.

The tournament uses profiles from the "Party" scenario of ANAC 2011 [14], available in Genius, where two friends together organize a party and negotiate about its location, type of music and more. This scenario is chosen for its diversity and its high outcome density, with 8 unique profiles and 3072 possible outcomes. To ensure meaningful negotiations, we selected the 25% most contrasting profile combinations by running a test tournament between two linear conceder agents ($m = 0, e = 1$) and identifying those with the highest agreement times. This prevents situations where agents' preferences align (almost) completely, allowing agreements to be reached too quickly and bypassing the negotiation process.

All negotiations are run with a deadline of 100, where we evaluate TSF methods by presenting cut off negotiations, that is a subset of the bidding rounds (10, 30, 50, or 70 data points). To ensure meaningful analyses, we exclude data instances (cut off negotiations) characterized by constant bidding curves with only repeated bids, as these cases do not exhibit any trend at all. Our TSF methods generate predictions based on parts of complete negotiations, enabling a comparison between the predicted outcome and the actual true outcome value. Given this, our chosen metric evaluating the accuracy must compare a point (the actual value) to a distribution (the forecast). First introduced by Matheson and Winkler [21], CRPS quantifies the difference between the perfect distribution of data

[4] Based on `https://github.com/Shengbo-Chang/BCI-opponent-model` (S. Chang).

$$CRPS = \int_{-\infty}^{\infty} (F(x) - \mathbb{1}\{y \leq x\})^2 dx$$

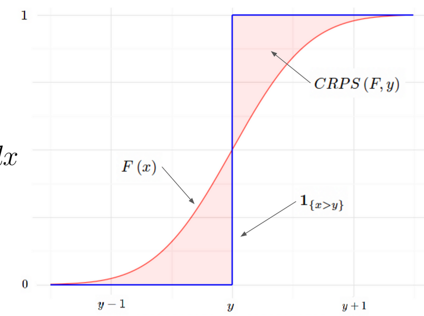

Figure 4: A visualization of the CRPS metric, adapted from [11].

(a pointwise distribution) and the predicted distribution, visualized in Figure 4, where $F(x)$ is the CDF of the predicted distribution and y the actual outcome value. Intuitively, the CRPS score can be interpreted as the distributive version of the Mean Average Error, describing the distance between the distribution mass and the true outcome, which we aim to minimize.

Evaluated TSF Methods. Firstly, we adapt the *Gaussian* utility prediction module by [29] for the agent IAMHaggler2011 [30]. The IAMHaggler2011 agent uses Gaussian process regression with a Matérn covariance function and a linear mean function to predict the opponent's utility curve. As input, they use the maximum utility of incoming bids within a small time window to minimize the effect of noise. As the agent is tailored to real-time negotiation, we adapt the module to suit our round-based negotiation setting. We convert round-based bids to a relative timescale and apply time windows accordingly. We also adapt the module to predict the agent's own curve as well. As the agent itself makes concessions and thus shows decreasing utility over time, we take the minimum of the agent's own bids when applying the method to the agent's own curve. The original agent is designed for the Genius platform [20] in Java; we use the Python implementation provided by NegoLog [10].

Secondly, we evaluate the performance of *ARIMA*, a widely used approach to TSF that describes autocorrelations in data [16] and produces distributional predictions. ARIMA, which stands for Auto-Regressive Integrated Moving Average, fits a model to the data based on three parameters: p, the lag order or the number of lag observations included in the model; d, the degree of differencing or the number of times the time series must be differenced to become stationary; and q, the order of the moving average or the size of the moving average window.

Finally, we use a naive benchmark method based on the intuition that both agents concede equally throughout the negotiation. This method *in between* estimates the final outcome as the midpoint between the utility of the opponent's initial bid and the agent's own initial bid. Note that the prediction of in between always corresponds to an agreement (no breakoff), so it is only used for outcome utility predictions, not as benchmark for the agreement probability.

4.2 Prediction Balance

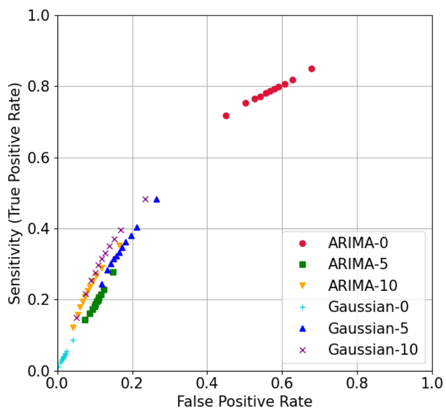

Figure 5: ROC curves of all aggregated negotiations for ARIMA and Gaussian with three different window sizes.

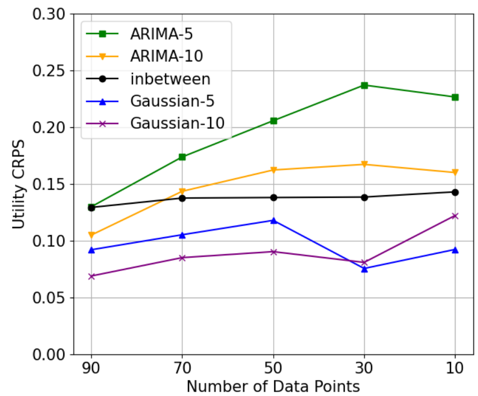

Figure 6: CRPS results for all aggregated negotiations with a decreasing number of data points. The x-axis is inverted to reflect increasing difficulty.

ARIMA and Gaussian both produce a distribution over the given input bid sequence, allowing Monte Carlo sampling to produce an outcome distribution and an agreement probability in the range 0 to 1. We introduce a threshold for the agreement probability, which enables us to classify negotiations as either a *breakoff* (positive classification) or an *agreement* (negative classification). Increasing the threshold raises the number of correctly identified breakoffs (true positives), but it also leads to more negotiations being incorrectly classified as breakoffs (false positives) and reduces the number of correctly identified agreements (true negatives). The ideal threshold therefore depends on the context and purpose of the prediction. The ROC curve shows the trade-off between the false positive rate (chance of false alarm) which we aim to minimize, and sensitivity (chance of a hit) which we aim to maximize.

In the ROC curve in Figure 5 visualized for all aggregated negotiations, we observe the difference between the TSF methods with different window sizes, where window size 0 means no window at all. While ARIMA with no window (0) shows a high sensitivity, this comes at a cost: the false positive rate is high, caused by too many predicted breakoffs. A window added to the algorithm decreases the noise in the input bidding sequence, allowing ARIMA to identify a clearer upward trend, and predict more agreements. On the other hand, Gaussian processes with no window predict almost no breakoffs, with a very low associated sensitivity ($<$0.1). When adding a window, the number of correctly identified breakoffs increase, showing the positive effect of the window size.

However, what characterizes as the most useful predictor depends on both the negotiation setting and the agent designer's goal. Overall, Gaussian with a window size of 10 outperforms the others across all negotiations when the desired false positive rate is around 15%. This effect is even more pronounced when we focus on time-dependent agent negotiations. In negotiations with ANAC agents only, Gaussian with window size 5 performs slightly better when aiming for 15% false positive rate, though the differences are small. Accordingly, the rest of the experiments we focus on ARIMA and Gaussian with a window size of 5 and 10.

4.3 Early Predictions

Outcome prediction in ongoing negotiations saves time and effort by ending unpromising negotiations. Earlier predictions, though harder, save the most time and thus are valuable. Figure 6 shows the trend between the number of data points — the number of rounds with associated bidding utilities presented to the TSF method — and the average CRPS of utility. We see that a smaller number of data points generally results in a worse result, i.e., a larger error from the real agreement utility (utility CRPS). However, early predictions should not only be characterized by the number of data points. For example, a negotiation cut off for prediction after 10 rounds with an agreement at 12 has few data points (10) but only requires 2 rounds of looking ahead, which is easier than a negotiation cut off at 10 but lasting much longer, e.g., until round 63. This is clearly visible when looking at a small number of data points (10) and a large number of rounds until agreement (>70) compared to a small number of rounds until agreement (<20). In this case, Gaussian with window size 5 yields an average CRPS of 0.13 (>70) and 0.05 (<20), performing better than the benchmark in between, showing the same effect with an average CRPS of 0.17 (>70) and 0.10 (<20).

4.4 Negotiator Types

The negotiations included in the tournament data can be split into categories of agent types: Time dependent agents with different opponent models (CUHK and HH), time dependent agents with simulated opponent models (0.25, 0.5), and complex, behavior-based agents (ANAC). Negotiations that show more trend, i.e., when the associated agents learned better with more advanced opponent models, would be expected to be easier to predict. Table 1 shows the average

	ANAC	TDA	TDA (.25)	TDA (.5)	TDA CUHK	TDA HH
Gaussian -5	0.156	0.090	0.090	0.086	0.133	0.119
Gaussian -10	**0.137**	**0.078**	**0.076**	**0.073**	0.115	**0.094**
ARIMA -5	0.190	0.171	0.169	0.160	0.242	0.293
ARIMA -10	0.169	0.128	0.132	0.117	0.223	0.183
inbetween	0.191	0.146	0.157	0.151	**0.112**	0.107

Table 1: The average (utility) CRPS for different methods and agent types.

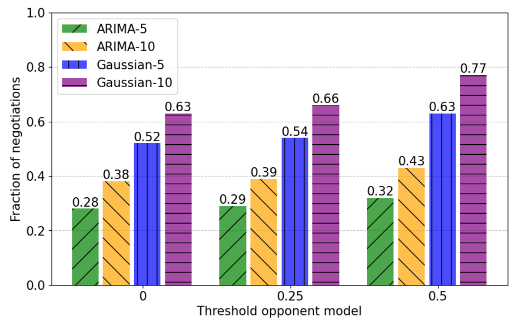

Figure 7: The percentage of negotiations where the real outcome has a ranking below 15%, shown for different opponent model thresholds.

(utility) CRPS categorized for different types of agent negotiations. As one can see, time dependent agents with a static opponent model show low CRPS values across all methods, which is surpassed in the negotiation with time dependent agents extended with simulated opponent models (0.25 and 0.5).

However, the CRPS values of negotiations between TDA with CUHK and HardHeaded opponent models do not show this improvement; instead, all methods (except for in between) show worse results compared to other TDA agent negotiations. This could be due to how well the opponent models CUHK and Hardheaded perform. We would expect that the percentage of breakoffs between agents decreases when combined with an increasingly advanced opponent model, because they better estimate their opponent preferences to find middle ground. This effect is visible when comparing the percentage of breakoffs between TDA, TDA (0.25) and TDA (0.5) with 33%, 26% and 14% of breakoffs, respectively. However, the percentage of breakoffs is much higher for TDA with HardHeaded and TDA with CUHK, namely 53% and 64%, respectively. This suggests that the opponent modeling did not function properly, which could explain why the CRPS values of these types of agents show worse results.

Despite employing opponent models, ANAC agents exhibit worse results in terms of CRPS values. The percentage of breakoffs is only 15%, which indicates that the quality of opponent modeling might not cause the effect of higher CRPS values. Instead, one should note that the interactions of ANAC agents are more complex than TDA agents, as they are behavior based agents, reacting on the bids from the opponent. This increases the complexity of the bidding curves, yielding a harder prediction challenge and resulting in higher CRPS scores.

4.5 Specific Outcome Prediction

The last step of the PrONeg's prediction pipeline predicts the specific outcome probabilities based on the utility outcome prediction. Some agent strategies could benefit from a ranking of all potential outcomes in order of likeliness to determine what selection of outcomes one should focus on and choose from. Of all predictions that predicted 'agreement' for negotiations that ended in agreement, we test the ranking of the real agreement, and show this as a percentage of all 3072 outcomes. The opponent models with thresholds of 0.25 and 0.5 are used to filter out outcomes where the opponent's utility falls below these respective values. Figure 7 shows the percentage of negotiations where the real outcome has a ranking below 15%, for different opponent models over TSF methods ARIMA and Gaussian with window size 5 and 10. Gaussian with window size 10 performs best with an opponent model threshold of 0.5, predicting that the real outcome is in the most likely 15% of outcomes in 77% of all predicted negotiations. Even with threshold 0, this percentage is 63%. However, one can see the effect of a good opponent model, increasing the scores significantly. An advanced opponent model would be necessary to increase the percentage of negotiations even further or to decrease the percentage threshold of likely outcomes.

5 Discussion

This paper introduces a general pipeline for outcome prediction in ongoing negotiations called PrONeg. Outcome prediction allows agents to end negotiations early if there is little hope for a good ending. Moreover, agents can adjust their strategies in parallel negotiations based on the expectation what specific outcome the negotiation will have, for example by imposing budget constraints in one-to-many negotiations [13].

Despite these benefits, this research is the first attempt to predict the outcome of an ongoing automated negotiation. In Table 2, we provide a comparison between the challenges of previous negotiation prediction studies. One line of research explores predicting the outcome in human-to-human negotiations [26,28] and hybrid human-agent negotiations [8,22]. However, these types of negotiations rely heavily on human psychological processes for their outcome [4], while automated negotiations often strip the human aspects of a negotiation. An automated agent is less susceptible to the psychological effects of anchoring or perspective taking that often influence human decision-making [18]. Nevertheless, valuable techniques have been developed, for example Moosmayer et al. [26] used neural networks to predict human negotiation outcomes and study the relationship between reference points and outcomes of business-to-business price negotiations. Another important line of research in automated negotiations focuses on predicting offers in advance, e.g., the expected counteroffer [7,31]. The Gaussian method designed by [29] extends the prediction to the complete opponent's negotiation curve, allowing us to use their technique as time series forecasting module in PrONeg. Beyond predicting an opponent's offers, opponent modeling research also explores the estimation of agent strategies [6,9,19], a topic comprehensively reviewed in [2].

Trainable Forecast Methods. Our work opens up interesting directions to research further. The current experimental setup is the first step in understanding the use of TSF methods in outcome prediction. Though the results are promising, we strive for a better accuracy for application purposes, especially in recognizing breakoffs. We invite new research to explore other TSF methods for dynamical model selection, and broaden the scope to trainable algorithms as well. For example, [31] employ two deep learning-based approaches to predict one bid ahead, indicating of the possibilities in feeding machine learning algorithms the traces of bids made to predict future bids. We have begun exploring deep learning methods for outcome prediction, with promising initial results.

Tactical Guidance. Agent's strategies can use the outcome prediction method PrONeg to end negotiation early when needed. This method could be expanded to guide the agent's strategy and provide tactical information to adjust the current bidding strategy. However, this requires further opponent modeling, e.g., how the opponent is expected to react on our (hypothetical) bidding curve [3] and integrating this in the intersection step. Agents could test different hypothetical bidding curves, and choose what curve is associated with their most favorable outcome. If the prediction is lower than wished for, the agent can adapt its strategy accordingly and aim for more. If the probability of any agreement at all is below preferred level, the agent can adjust the strategy and concede more.

Negotiation Support. Connecting the realms of automated and human negotiations, there is a growing body of research on automated agents that negotiate with humans in natural language. In an attempt to improve such natural language bots, Chawla et al. [8] analyzed the language used in bilateral buyer-seller negotiations and used this data to train a prediction model (BERT), which attempts to predict the outcome of these human-to-human negotiations, and Mell et al. [22] use a machine learning model trained on detected emotions among other parameters based on the text messages that go back and forth. Future negotiation systems may be able to combine these techniques for human negotiations based on text interpretations with the current research on the (abstract) course of bidding to make more accurate predictions for agent-assisted negotiations and provide more advanced strategy recommendations.

Challenge	Research
Predicting Human-to-human Negotiation Outcomes	Moosmayer et al. [26], Van Poucke et al. [28]
Predicting Hybrid Negotiation Outcomes	Chawla et al. [8], Mell et al. [22]
Predicting Automated Negotiation Offers	Carbonneau et al. [7], Williams et al. [29], Yesevi et al. [31]
Predicting Automated Negotiation Strategies	Brzostowski et al. [6], Hou et al. [9], Li et al. [19]
Predicting Automated Negotiation Outcomes	This Research

Table 2: Comparison of selected negotiation prediction studies.

Acknowledgments. This publication is part of the Vidi project COMBINE (VI.Vidi.203.044) which is (partly) financed by the Dutch Research Council (NWO).

Disclosure of Interests. Authors have no competing interests to declare.

References

1. Baarslag, T., Elfrink, T., Mofakham, F., Koça, T., Kaisers, M., Aydogan, R.: Bargaining Chips: Coordinating One-to-Many Concurrent Composite Negotiations. pp. 390–397. Australia (2021). https://doi.org/10.1145/3486622.3494023
2. Baarslag, T., Hendrikx, M.J.C., Hindriks, K.V., Jonker, C.M.: Learning about the opponent in automated bilateral negotiation: a comprehensive survey of opponent modeling techniques. Autonomous Agents and Multi-Agent Systems **30**(5), 849–898 (2016). https://doi.org/10.1007/s10458-015-9309-1
3. Baarslag, T., Hindriks, K., Jonker, C.: Towards a Quantitative Concession-based Classification Method of Negotiation Strategies. pp. 143–158. Springer (2011). https://doi.org/10.1007/978-3-642-25044-6_13
4. Bazerman, M.H., Curhan, J.R., Moore, D.A., Valley, K.L.: Negotiation. Annual review of psychology **51**(1), 279–314 (2000)
5. Brazier, F.M.T., Cornelissen, F., Gustavsson, R., Jonker, C.M., Lindeberg, O., Polak, B., Treur, J.: A multi-agent system performing one-to-many negotiation for load balancing of electricity use. Electronic Commerce Research and Applications **1**(2), 208–224 (2002). https://doi.org/10.1016/S1567-4223(02)00013-3
6. Brzostowski, J., Kowalczyk, R.: Adaptive Negotiation with On-Line Prediction of Opponent Behaviour in Agent-Based Negotiations. In: IEEE/WIC/ACM International Conference on Intelligent Agent Technology. pp. 263–269. IEEE, Hong Kong, China (2006). https://doi.org/10.1109/IAT.2006.26
7. Carbonneau, R.A., Kersten, G.E., Vahidov, R.M.: Pairwise issue modeling for negotiation counteroffer prediction using neural networks. Decision Support Systems **50**(2), 449–459 (2011). https://doi.org/10.1016/j.dss.2010.11.002
8. Chawla, K., Lucas, G., May, J., Gratch, J.: Exploring Early Prediction of Buyer-Seller Negotiation Outcomes (2021). https://doi.org/10.48550/arXiv.2004.02363
9. Chongming Hou: Predicting agents tactics in automated negotiation. In: Proceedings of International Conference on Intelligent Agent Technology. pp. 127–133. IEEE, Beijing, China (2004). https://doi.org/10.1109/IAT.2004.1342934
10. Doğru, A., Keskin, M.O., Jonker, C.M., Baarslag, T., Aydoğan, R.: NegoLog: An Integrated Python-based Automated Negotiation Framework with Enhanced Assessment Components. In: Proceedings of the 33rd IJCAI. vol. 9, pp. 8640–8643 (2024). https://doi.org/10.24963/ijcai.2024/998
11. Faran, I.: CRPS - A Scoring Function for Bayesian Machine Learning Models (2023), https://towardsdatascience.com/crps-a-scoring-function-for-bayesian-machine-learning-models-dd55a7a337a8/
12. Faratin, P., Sierra, C., Jennings, N.R.: Negotiation decision functions for autonomous agents. Robotics and Autonomous Systems **24**(3-4), 159–182 (1998). https://doi.org/10.1016/S0921-8890(98)00029-3
13. Florijn, T.C.P., Yolum, P., Baarslag, T.: A Survey on One-To-Many Negotiation: A Taxonomy of Interdependency. In: Proceedings of the 34rd IJCAI (2025)

14. Fujita, K., Ito, T., Baarslag, T., Hindriks, K., Jonker, C., Kraus, S., Lin, R.: The Second Automated Negotiating Agents Competition (ANAC2011). In: Complex Automated Negotiations: Theories, Models, and Software Competitions, vol. 435, pp. 183–197. Springer (2013). https://doi.org/10.1007/978-3-642-30737-9_11
15. Hao, J., Leung, H.f.: CUHKAgent: An Adaptive Negotiation Strategy for Bilateral Negotiations over Multiple Items. In: Novel Insights in Agent-based Complex Automated Negotiation, Studies in Computational Intelligence, vol. 535, pp. 171–179. Springer, Tokyo (2014). https://doi.org/10.1007/978-4-431-54758-7_11
16. Hyndman, R.J., Athanasopoulos, G.: Forecasting: Principles and Practice (2018)
17. van Krimpen, T., Looije, D., Hajizadeh, S.: HardHeaded. In: Complex Automated Negotiations: Theories, Models, and Software Competitions, pp. 223–227. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-30737-9_17
18. Kristensen, H., Gärling, T.: The Effects of Anchor Points and Reference Points on Negotiation Process and Outcome. Organizational Behavior and Human Decision Processes **71**(1), 85–94 (1997). https://doi.org/10.1006/obhd.1997.2713
19. Li, M., Murukannaiah, P.K., Jonker, C.M.: A Data-Driven Method for Recognizing Automated Negotiation Strategies (2021). https://doi.org/10.48550/arXiv.2107.01496
20. Lin, R., Kraus, S., Baarslag, T., Tykhonov, D., Hindriks, K., Jonker, C.M.: Genius: An Integrated Environment for Supporting the Design of Generic Automated Negotiators. Computational Intelligence **30**(1), 48–70 (2014). https://doi.org/10.1111/j.1467-8640.2012.00463.x
21. Matheson, J.E., Winkler, R.L.: Scoring rules for continuous probability distributions. Management science **22**(10), 1087–1096 (1976)
22. Mell, J., Beissinger, M., Gratch, J.: An Expert-Model & Machine Learning Hybrid Approach to Predicting Human-Agent Negotiation Outcomes. J. Multimodal User Interfaces **15**(2), 215–227 (2021). https://doi.org/10.1007/s12193-021-00368-w
23. Mohammad, Y.: Concurrent local negotiations with a global utility function: a greedy approach. Autonomous Agents and Multi-Agent Systems **35**(2), 28 (2021). https://doi.org/10.1007/s10458-021-09512-y
24. Mohammad, Y., Nakadai, S., Greenwald, A.: NegMAS: A Platform for Automated Negotiations. pp. 343–351. Lecture Notes in Computer Science, Springer (2021). https://doi.org/10.1007/978-3-030-69322-0_23
25. Mohammad, Y., Viqueira, E.A., Ayerza, N.A., Greenwald, A., Nakadai, S., Morinaga, S.: Supply Chain Management World: A Benchmark Environment for Situated Negotiations. In: PRIMA, vol. 11873, pp. 153–169. Springer (2019). https://doi.org/10.1007/978-3-030-33792-6_10
26. Moosmayer, D.C., Chong, A.Y.L., Liu, M.J., Schuppar, B.: A Neural Network Approach to Predicting Price Negotiation Outcomes in Business-To-Business Contexts. Expert Systems with Applications **40**(8), 3028–3035 (2013). https://doi.org/10.1016/j.eswa.2012.12.018
27. Rubinstein, A.: Perfect Equilibrium in a Bargaining Model. Econometrica: J. Econometric Society **50**(1), 97–109 (1982), https://doi.org/10.2307/1912531
28. Van Poucke, D., Buelens, M.: Predicting the Outcome of a Two-Party Price Negotiation: Contribution of Reservation Price, Aspiration Price and Opening Offer. J. Economic Psychology **23**(1), 67–76 (2002). https://doi.org/10.1016/S0167-4870(01)00068-X
29. Williams, C.R., Robu, V., Gerding, E.H., Jennings, N.R.: Using Gaussian Processes to Optimise Concession in Complex Negotiations Against Unknown Opponents. pp. 432–438 (2011)

30. Williams, C.R., Robu, V., Gerding, E.H., Jennings, N.R.: IAMhaggler2011: A Gaussian Process Regression Based Negotiation Agent. In: Complex Automated Negotiations: Theories, Models, and Software Competitions, pp. 209–212. St. Computational Intelligence, Springer (2013). https://doi.org/10.1007/978-3-642-30737-9_14
31. Yesevi, G., Keskin, M.O., Doğru, A., Aydoğan, R.: Time Series Predictive Models for Opponent Behavior Modeling in Bilateral Negotiations. In: PRIMA, vol. 13753, pp. 381–398. Springer (2023). https://doi.org/10.1007/978-3-031-21203-1_23