A Negotiation Model for On-line Learning

D. (Didi) Rustam, Th.P. (Theo) van der Weide

Version of July 11, 2018 at 13:19

Abstract

The concept of e-Learning is rapidly getting worldwide recognition with massive on-line courses (MOOCs) as a major exponent. In this paper, we focus on a knowledge market for selling of on-line courses.

1. We describe the existing approach and define the knowledge market in our context. In this paper specifically about package selection and negotiation.
2. Then we discuss more detail about the products generation, a series of courses that valid for a student to achieve his course program can be generated from the market. From that packages, the student may want to find series that fit to his preferences
3. Educational institutions as the course provider may want to automate the selling process including the negotiation. We focus on a single course negotiation within a series of courses.
4. We test our proposed by doing experiments in such a higher education in Indonesia system, and present some results.
5. Closed with some conclusion based-on our finding.

Keywords: negotiation model, bargaining, on-line learning

1 Introduction

In this paper, we describe part of our approach (see [1, 2]) to design the matching system for the student who are looking for the e-learning course in a e-market platform, and develop automatic bargaining mechanism to help seller handling a huge number of users.

Our target application is a situation where higher education institutions (HEI's) offer on-line course programs to candidate students. Students want to do a teaching program for which they have to complete courses. Students can select courses from the of the HEI's offers. The HEI's try to make a profit by selling their courses. The students have some budget for their study program and try to find the best courses for the best price such that they stay within their budget.

The virtual knowledge service market is a place where buyer and seller to meet to exchange their supplies and demands [1]. From seller perspective, the process consists of the selection of prospective students to follow their course, which among others involves taking into account the reputation ratings from students, and the calculation of the amount of profit for selling a course. From buyer perspective, the model captures the selection process of a coherent set of courses to realize their educational goals. The negotiation about individual courses typically involves the course reputation, the reputation of the course provider (seller), and the price of a course.

Products of the market can be a single course or a course-package. To achieve their goal, a student needs to complete the courses or a course-package. The product has some attributes, such as Pre- and Post-condition which is student expertise before and after they complete a course. Other attributes are price and reputation.

Any institution may offer a course in the market. The institution may have a different reputation. Hence, although a course is having the same of pre- and post-condition but the preference of each student to price and reputation could have been different. The reputation of the institution may affect the value of course.

The market is equipped with a course-package generator. This generator will generate a course-package based on the Pre- and Post-condition of student and course. The generator is also matching the preference of student and seller; reputation and price.

In an on-line learning market, a huge number of buyers (students) can be expected. Rather than having fixed prices, we assume a more flexible pricing mechanism that also takes typical educational values into account. A consequence is that a negotiation process is required to find an agreement on the price. Individual buyers may want to do the negotiations themselves, but a higher education institution may prefer to have negotiating agents. This institution requires a mechanism making the negotiation process run smoothly, and thus not burdening the course provider (seller).

Buyers may have very different backgrounds. For example, buyers may come from universities that are not popular, will have different levels of knowledge and skills, and will have a different budget for buying courses.

There are several forms of negotiation: bidding, auction, and bargaining. The simplest form of negotiation is bidding, in which a buyer specifies the products required, and then asks the suppliers to make a bid. Then the buyer selects the supplier with the best offer. An auction is a form of negotiation in which buyers compete for a restricted set of products. The auctions follow a strict, fixed auction protocol. Bargaining is the most complex form of negotiation, in which offers and counteroffers are exchanged between the negotiating parties until they reach an agreement or disagreement. In this paper, we focus on bargaining.

The rest of the article is organized as follows. We discuss about the existing approach in Section 2. We describe the context of the paper in Section 3. In Section 4 and Section 5 we describe the product generation and negotiation model, respectively. We present some results based on the experiments in Section 6. Finally, in Section 7 we draw some conclusions.

2 Literature Review

Some researchers studies about the e-learning market specifically in higher education. For example [3] reviews the role of US higher education in the global e-learning market, [4] analyze the issues and prospects of e-learning markets and providers, and [5] propose a digital marketplace for education.

In a traditional market, individual consumers and sellers usually want to negotiate the price, quality of goods, delivery date, quantity of products, or other purchase condition. The negotiation normally is conducted by people involved in business transaction.

In an electronic marketplace, the negotiation is likely to be processed either automatically or at least semi-automatically with as less as possible human interaction. Automated negotiation research are available recently [6, 7, 8].

3 The Knowledge Market Context

This study is situated in the context of Indonesia Higher Education on-line learning and its particular educational system. In this section, this context is explained. The focus here is on the negotiation of on-line course programs that take place between students and institutions as the course provider.

As proposed in our previous paper (see [2]), on-line learning knowledge market is a type of closed marketplace where students (buyers) find the course(s) which provided by higher education institutions (sellers). Technically, knowledge market is an IT-Supported platform that helps student queries to available and appropriate courses or series-of-courses. A closed marketplace in the sense of course-quality is watched carefully.

Although the market is closed, each university can offer their courses in the market. Thus, the courses that are available may vary in term of categories and types. While students may also consist of different characters, capability (reputation), preferences, and budget.

In this section, we will explain how the proposed knowledge market works. 1) Facilitating a prospective student find a suitable packages of courses that match to his preferences and budgets. 2) Helping seller to handle numbers of buyers and their bargaining automatically.

In this paper we will specifically discuss about Digital market as shown in Figure 1

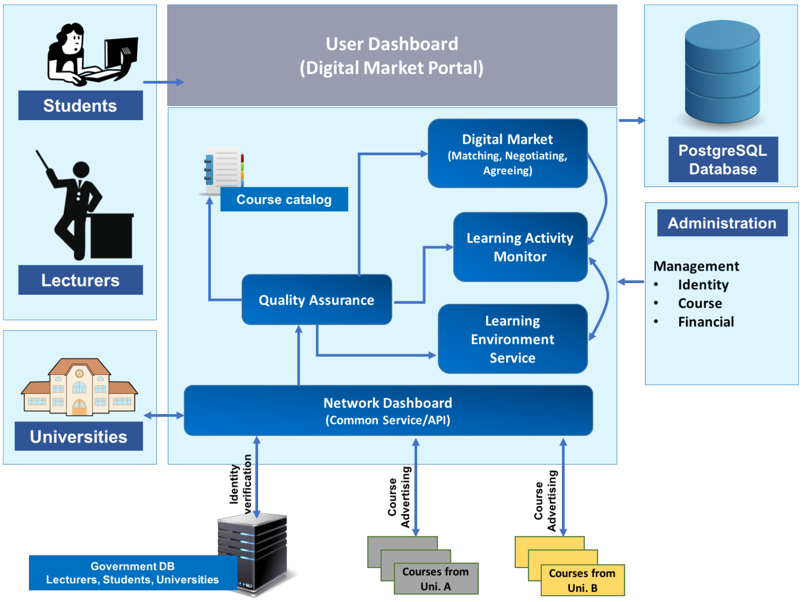

Figure 1: Overview

3.1 Product Packages Generation

Course as main products of our market. Products of the market not only a simple product (single course) but may also in form of course series. Figure 2 shown an example of products packages. For example Course A has condition Pre=1 and Post=5, while Course C has Pre=5 and Post=7. Course B with condition Pre=1 and Post=7.

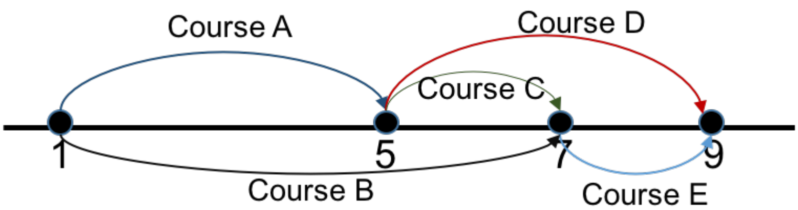

Figure 2: Course Package

By modeling this way, we can do matching the required courses by a student to complete his goal. For example, suppose two students named StA and StB. Both are having the same goals, want to achieve 1 to 7. Using our model, both may take a package of course consists of CourseA, CourseC or a package of course with single CourseB.

We may define products packages is a package that consists of a single course or series of courses that valid for a student to achieve his goal.

User can generate products packages that fit to his course program for achieving his goal. This process can be done by generating the package of course(s) from LMS. In this paper, we will specifically discuss Moodle; one of a free and open-source LMS that widely used specifically in Indonesia nowadays [9, 10].

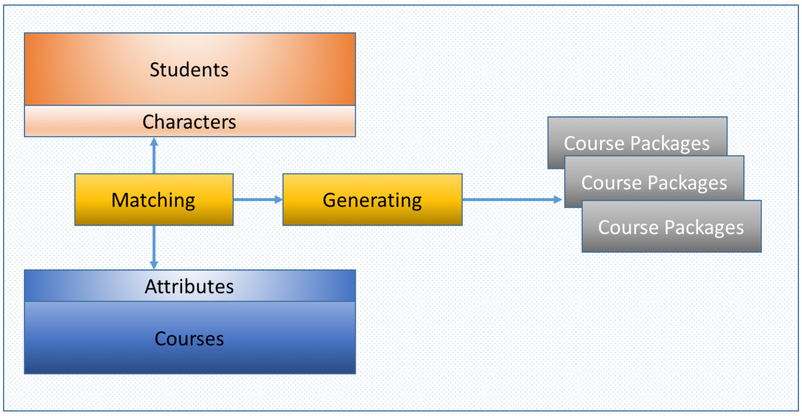

Figure 3: Package Generation

The Modular Object-Oriented Dynamic Learning Environment (Moodle) is free and open-source and distributed under the GNU General Public License. For a short introduction to Moodle, see [11]. A large benefit of open-source software is that the code is open for scrutiny. Consequently, it is supported by a global developer community accessing and modifying the code, leading to a more stable and secure software package. Moodle is configurable, highly-flexible and feature-reach. With over 500 Moodle plugins, it can be modified to fit most user's need.

Our course assembler will pull the required data from Moodle, as shown in our conceptual model (see Figure 6). This information is restricted to the the internal Moodle conceptual structure. However, it requires some extensions in the Moodle. Since courses in Moodle do not have attributes as Price, Pre, and Post. This extension of have to be handled by the negotiation module itself.

3.2 Negotiation

When discussing negotiation, it is relevant to distinguish between negotiation strategy and negotiation protocol. Negotiation strategy is the way in which a given party acts within those rules in an effort to get the best outcome of the negotiation. The strategy of each participant is private. While negotiation protocol is the flow of messages between the negotiation parties, guiding the negotiation parties to get some actions in the interaction. The protocol is necessarily public and open.

As noted above, in this paper we focus on bargaining negotiation. In our case, we introduce two bargaining strategies; defensive and aggressive. An aggressive buyer makes small re-bargaining steps in each round, while a defensive seller makes large steps. Defensive buyers and aggressive sellers are doing just the opposite. Low reputation buyers will tend to be defensive, since they expect the buyer not to make many concessions to get them as a customer. On the other hand, high reputation buyers *(mutatis mutandis)* will tend to be aggressive. The same style of reasoning can be used for the sellers.

We employ the communication primitives as the negotiation protocol in our negotiation system (see Table 1).

Propose	Send a proposal or a counter-proposal
Accept	Accept the terms and conditions specified in a proposal without further modifications
Terminate	End current negotiation process and terminate session
Reject	Reject the current proposal with or without an attached explanation
Acknowledge	Acknowledgement after receiving a message
Modify	Modification to the proposal sent last
Withdraw	Withdraw the latest proposal
CFP	Client to initiate a call-for-proposal

Table 1: Negotiation Primitives

As shown in Figure 4, sellers advertise their products and buyers describe their demands.

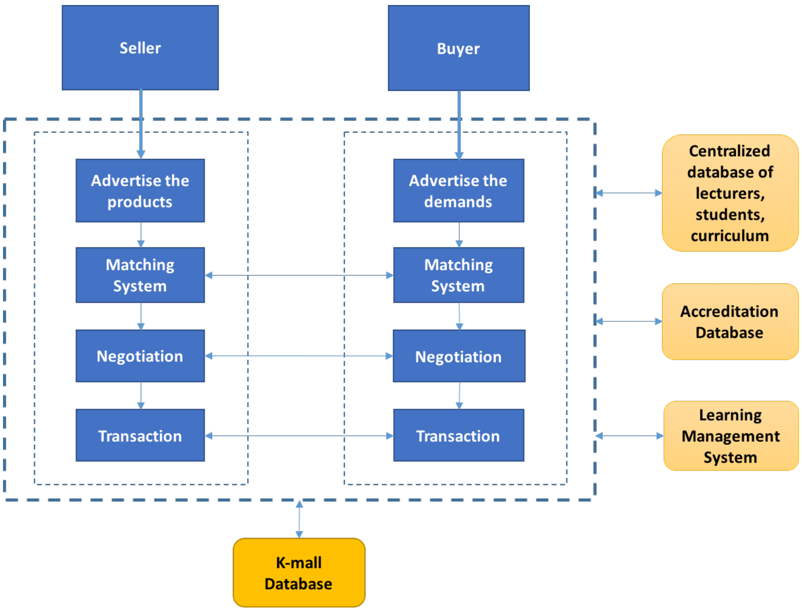

Figure 4: Negotiation architecture

This information basically fills the database with the relational scheme described in see Figure 6. The matching systems couples demand and supply, and helps the seller to find interested (potential) buyers, and helps buyers to find the products they need. In our case, the buyers (students) need combinations of products that can satisfy their (educational) needs. The matching system will help to find the possible combinations. Once the possible matches are found, both parties can have agreement when they can agree upon the price. In this paper we assume that products (courses) are negotiated in isolation. A more advaced bargaining mechanism, that involves negotiations of product packages, is outside the scope of this paper. After reaching agreement, buyer and seller can go to next phase, which is the transaction.

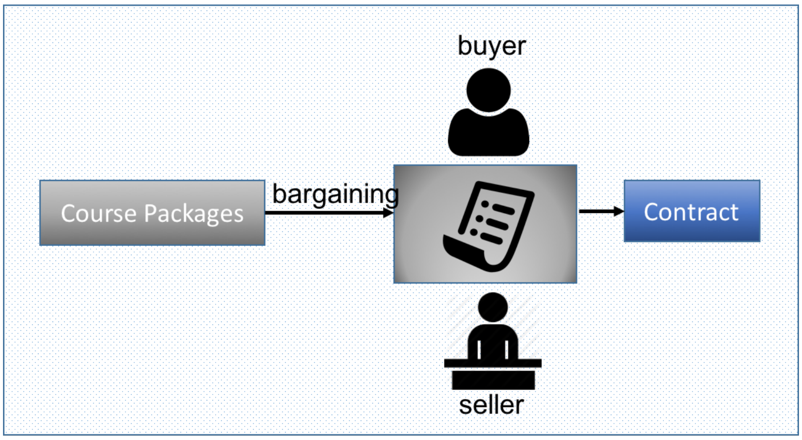

Figure 5: Course Negotiation

The negotiation model is based on sellers (educational institutions) offering products (courses) to buyers (students). The courses are offered via a digital market. The underlying conceptual model is show an a relational model in Figure 6. This model is a simplified version of the model presented in [1], focusing on the attributes that are essential to describe the bargaining model.

Courses are identified by their name and the selling agency. In this paper we restrict ourselves to the following course attributes: 1. the advertised price of the course, 2. the required pre-knowledge to enter the course, and 3. the knowledge level obtained after successfully completing course.

Sellers are the education institutions. They are described by the relational scheme Seller, see Figure 6.

The utility function for buyers and sellers is derived from the valuation of the attributes. For example, a buyer may find a successful student more attractive, then this is reflected in the utility function. The individual scores are aggregated according to the Analytic Hierarchy Process (AHP) ([12]).

In the negotiation process buyer and seller take turns in which the seller makes a (next) offer for the product under negotiation and the buyer does a (next) bid. The next step of both seller and buyer depends on the how they evaluate the benefit of the possible transaction, and how their strategy is to overcome the difference in price offer and bid.

4 Products Generation

In this section we focus on the generation of candidate course packages. The notation Cand(P) indicates that P is a candidate package. We describe an inductive definition of candidate packages, and use this definition to derive a generation algorithm.

Buyers are described in this paper by the following attributes: 1. the current knowledge state, 2. the intended knowledge state, and 3. the buyer budget.

Course(Name, Seller, Price, Pre, Post)

Seller(Name, Rep)

Buyer(Name, Budget, Pre, Post)

Figure 6: The relation scheme

We will consider sequences of courses, and use the following notational conventions. Let C be a course sequence, then $n(C)$ is the length of this sequence (number of subsequent courses), C_i the i-th course in this sequence ($1 \leq i \leq n(C)$), and $C\,[i{:}j]$ the (possibly empty) subsequence of courses from position i up to position j. If C and D are sequences of courses, then $C{+}D$ is the sequence of courses resulting from concatenating D to C. When no confusion is likely to occur, we will use c as denotation for the single length course sequence consisting of course c only.

4.1 Admissible course sequences and packages

We call a course sequence C an admissible sequence of courses, denoted as: $\mathsf{Admiss}(C)$, if for each each of its courses guarantees admissibility to the next course:

Definition 1 $\mathsf{Admiss}(C) \triangleq \forall_{1 \leq i < n(C)}\,[C_i.post \geq C_{i+1}.pre]$

An admissible sequence C of courses is called a package, denoted as $\mathsf{Pack}(C)$, if it does not contain any superfluous courses, meaning that after removing a intermediate course from a package the result is not an admissible sequence.

4.1.1 Inductive description of packages

We will be interested in an inductive description of packages. This inductive description will be the basis of the inductive generation of packages that we will describe later. Single courses form the basis for the inductive description.

Lemma 1 Let c be a course, then c also is a (singleton) package.

Next we consider the extension $c{+}C$ of a package C with a next course c. Then the result is an admissible sequence of courses, or $\mathsf{Admiss}(c{+}C)$, if $c.post \geq C_1.pre$. Would $c{+}C$ contain superfluous courses, then there are the following 2 cases:

1. Course c is superfluous in $c{+}C$.

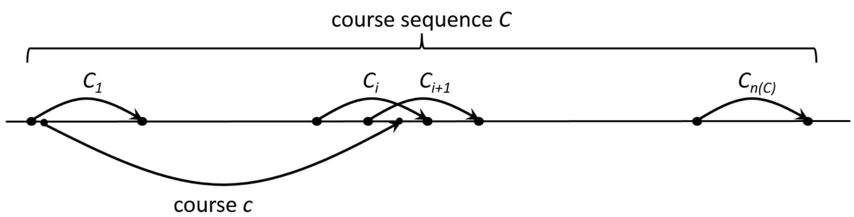

Figure 7: Course c superfluous

This happens when course c is subsumed by an initial segment of package C. A necessary and sufficient condition for this situation is: $c.pre \geq C_1.pre \wedge c.post \leq C_{n(C)}.post$. So the extension requirement to avoid this case is $c.pre < C_1.pre \vee c.post \leq C_{n(C)}.post$. We will strengthen this condition since we will focus on extension by prepension only, into:

$$c.pre < C_1.pre$$

2. An initial segment $C\,[1{:}i]$ of package C for some $1 \leq i \leq n(C)$ is superfluous in $c + C$.

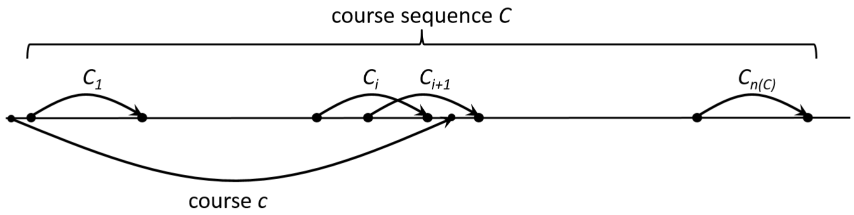

Figure 8: Initial segment superfluous

In this case especially course C_1 is superfluous in $c{+}C$. A necessary and sufficient condition for this case is: $c.pre < C_1.pre \wedge c.post \geq C_2.pre$. So the extension requirement to avoid this this case is:

$$c.pre \geq C_1.pre \vee c.post < C_1.post$$

We conclude:

Lemma 2 Packages are inductively generated as follows:

1. If c is a course, then $\mathsf{Pack}(c)$.

2. If $\mathsf{Pack}(C)$ and both $c.post < C_1.pre$ and $c.pre \geq C_1.pre \vee c.post < C_1.post$ then also $\mathsf{Pack}(c{+}C)$.

4.2 Candidate packages

Next we focus on the selection of suitable candidate packages for buyers (students). A package C is valid for buyer B as an educational program for buyer Bs educational needs if both the admission requirement $C_1.pre \leq B.pre$ and the goal $B.post \leq C_{n(C)}.post$ of the buyer are satisfied.

Definition 2

$$\begin{aligned}\mathsf{Valid}(C, B) \triangleq\ & \mathsf{Pack}(C) \\ & \wedge B.pre \geq C_1.pre \wedge C_{n(C)}.post \geq B.post\end{aligned}$$

A candidate package for buyer B is a package that is valid for that buyer and a minimal program to achieve that educational goal. First we note that each singleton package that is valid for buyer B obviously also is a candidate package for that buyer. Next we consider package P with length $n = n(P)$ larger than 1. Then P is a candidate package for buyer B, denoted as $\mathsf{Cand}(C, B)$, if:

1. package P is a valid package for buyer B and thus:

$$B.pre \geq C_1.pre \wedge C_n.post \geq B.post \tag{1}$$

2. C_n is required to achieve buyer's goal. So package $C[1{:}n-1]$ is not a valid package for buyer B, or $\neg\mathsf{Valid}(C[1{:}n-1], B)$. Thus:

$$\neg(B.pre \geq C_1.pre \wedge C_{n-1}.post \geq B.post) \tag{2}$$

3. C_1 is required to be an admissible package. So package $C[2{:}n]$ is not a valid package for buyer B, or $\neg\mathsf{Valid}(C[2{:}n], B)$. Thus:

$$\neg(B.pre \geq C_2.pre \wedge C_n.post \geq B.post) \tag{3}$$

Condition 2 and the first part of condition 1 are combined as follows:

$$\begin{aligned}&\neg(B.pre \geq C_1.pre \wedge C_{n-1}.post \geq B.post) \wedge B.pre \geq C_1.pre\\&= C_{n-1}.post < B.post \wedge B.pre \geq C_1.pre\end{aligned}$$

Condition 3 and the second part of condition 1 lead to:

$$\begin{aligned}&\neg(B.pre \geq C_2.pre \wedge C_n.post \geq B.post) \wedge C_n.post \geq B.post\\&= B.pre < C_2.pre \wedge C_n.post \geq B.post\end{aligned}$$

Summarizing we get:

Lemma 3 $\mathsf{Cand}(C, B) \Leftrightarrow \mathsf{Valid}(C, B)$
$\wedge\ (C_1.pre \leq B.pre < C_2.pre$
$\wedge\ C_{n-1}.post < B.post \leq C_n.post)$

Next we extend the generation of packages to the generation of candidates. A package P is called a semi-candidate for buyer B if it has no superfluous courses and meets the eduction goal of buyer B:

Definition 3 $\mathsf{SCand}(P, B) \triangleq \mathsf{Pack}(P) \wedge B.post \leq P_{n(P)}.post$

Then a semi-package is complete when it is admissible for the associated buyer:

Lemma 4 $\mathsf{SCand}(P, B) \wedge P_1.pre \leq B.pre \Rightarrow \mathsf{Cand}(P, B)$

Semi-candidates are generated by the following inductive definition:

Lemma 5 Semi-candidates are inductively generated as follows:

1. If c is a course and $B.post \leq c.post$, then $\mathsf{SCand}(c)$.

2. If $\mathsf{SCand}(P, B)$ and $P_1.pre > B.pre$ then $\mathsf{SCand}(c + P)$.

So, being a semi-candidate can be seen as the invariant relation for our candidate generation algorithm.

5 Negotiation Model

A most simple bargaining model would be an iterative process where p_s^i is the price offered by the seller in the i-th round, and p_b^i the price bid by the buyer in this round. Initially we have: $p_s^0 = C.\mathsf{Price}$ and $p_b^0 = 0$. The i-th ($i > 0$) iteration step is described by:

1. the buyer makes a bid of $p_b^i = p_b^{i-1} + \mathsf{Step}_b(C, s, p_s^{i-1})$ for product C to seller s, where $\mathsf{Step}_b(C, s, p_s^{i-1})$ depends on the quality of the product, the quality of the seller and the reputation of the seller.

2. the seller makes an offer $p_s^i = p_s^{i-1} - \mathsf{Step}_s(C, s, p_b^{i-1})$ using the seller-specific function $\mathsf{Step}_s(C, b, p_b^{i-1})$.

The negotiation stops when after some iteration the price offered meets the bid, or when an offer or bid is accepted or when either seller or buyer looses interest.

Next we describe the algorithms we used in our bargaining system. To compute buyer utility, we used Algorithm **??**. First, buyer should input the weight for each criteria; price, reputation of course seller, and reputation of the course it self. We use weight value to see the preference of buyer, and to observe which criteria more important than other. From this weight value, and criteria values taken from the database, the buyer utility can be calculated by compute the average of weight for each criteria. Normalize price is required, since the value can be more than 1. We normalize the price to the highest price of available courses. Note that all attribute values are assumed to be normalized to the interval $[0, 1]$. Furthermore, the weights may be obtained by the strategy proposed by the Analytic Hierarchy Process (AHP) ([12]) as described in Section **??**.

In this system, we propose two mechanism of bargaining, which are manual (semi-automatic) and fully automatic. Our aim is to avoid waiting time and to help seller to handle the number of buyer (potential student will enroll to the course) in bargaining stage.

First we discuss the contribution of the buyer who we assume to prefer to bargain manually. This is shown in Algorithm 1. The buyer first checks whether the proposed price from the seller involved for the course under negotiation is acceptable, by calling *acceptableB (seller, price, bid).* This will involve a utility computation by the buyer. If no agreement, then the buyer first decides to abort the bidding process or not. If not, an new bid is entered (manually) by the buyer, and the bargaining process is continued with the a call to the bargaining function *Sneg* of the seller.

Next we discuss Algorithm 1 that handles the automatic bargaining of the seller. In this case, the new offer is computed rather than entered by the seller.

In the next section we focus on the generation of the candidate course packages that can satisfy the educational needs of the students.

Algorithm 1: Buyer manual bargaining

```
output: agreed price or failure
def Bneg(course, seller, price, buyer, bid, nr):
    // course is offered by seller for price;
    // bid is most recent bid of buyer;
    // nr = Maximum number of remaining iterations;
    if acceptableB(seller, price, bid):
        return deal;
    elif nr > 0 and continuableB(seller, price, bid, nrBids):
        newbid = ask new buyer bid;
        return Sneg(course, seller, price, buyer, newbid, nr-1);
        else:
            return failure;
```

Algorithm 2: Seller automatic bargaining

```
output: deal price
def Sneg(course, seller, price, buyer, bid, nr):
    if acceptableS(price, buyer, bid):
        return deal;
    elif nr > 0 and continuableS(price, buyer, bid, nr):
        newprice = newOffer(price, buyer, bid, nr);
        return Bneg(course, seller, newprice, buyer, bid, nr-1);
        else:
            return failure;
```

6 Experiments

In this section we first describe the digital learning management system (LMS) that may adopt our proposed course negotiation model. Then we describe the major structure of the implementation. We continue with the generation of candidate course packages for buyers and then describe the implementation in terms of a relational database system.

We will describe the generation of candidate packages for buyers via a recursive query formulated in PostgreSQL ([13]). This leads to the following generation algorithm:

```
WITH RECURSIVE
  packages (Bid, Bname, Bpre, Bpost,
            pre, step1, step2, post, courses, price) as (
    SELECT B.id, B.name, B.pre, B.post,
           C.pre, C.post, C.post, C.post, ARRAY[C.id],
           C.price
    FROM   courses C, buyers B
    WHERE  C.post >= B.post AND C.pre < B.post
  UNION ALL
    SELECT P.Bid, P.Bname, P.Bpre, P. Bpost,
           C.pre, C.post, P.pre, P.post, C.id || P.courses,
           C.price + P.price
    FROM   courses C, packages P
    WHERE  P.pre  >  P.Bpre  -- semi-candidate not complete?
      AND  C.post >= P.pre   -- requirements fulfilled?
      AND  C.pre  <  P.pre   -- improving previous course?
      AND  C.post <  P.step2 -- advances the level?
  )
INSERT INTO candidates
  SELECT Bname, Bpre, Bpost, pre, post, courses, price
  FROM   packages
  WHERE  pre <= Bpre;
```

7 Conclusion and Future Work

In this paper we described an implementation model for an automated negotiation model in the virtual knowledge market model. Our next steps are to extend the prototype in order to tune the system parameters and to demonstrate that the proposed model can be an effective solution in general, but in particular in the situation of Indonesia.

Acknowledgement

The authors would like to thank to Ministry of Research, Technology and Higher Education Indonesia (RISTEKDIKTI) which has been funded this research.

References

[1] D. Rustam, T. van der Weide, Towards a methodological framework for designing a knowledge market, in: Advanced Informatics: Concept, Theory and Application (ICAICTA), 2014 International Conference of, 2014, pp. 232–237. doi:10.1109/ICAICTA.2014.7005946.

[2] D. Rustam, T. van der Weide, Virtual knowledge service market: A framework for higher education online learning, in: Business Informatics (CBI), 2015 IEEE 17th Conference on, Vol. 2, 2015, pp. 55–56. doi:10.1109/CBI.2015.25.

[3] M. van der Wende, The rise of the knowledge market (January 2002) [cited 10 May 2016].
URL http://www.cshe.berkeley.edu/sites/default/files/shared/publications/docs/ROP.WendePaper1.02.pdf

[4] K. Fry, E-learning markets and providers: some issues and prospects, Education + Training 43 (4/5) (2001) 233–239. arXiv:http://dx.doi.org/10.1108/EUM0000000005484, doi:10.1108/EUM0000000005484.
URL http://dx.doi.org/10.1108/EUM0000000005484

[5] B. Woolf, V. Lesser, C. Eliot, Z. Eyler-Walker, M. Klein, A digital marketplace for education, in: W. Chin, F. Patricelli, V. Milutinovi (Eds.), Electronic Business and Education, Vol. 20 of Multimedia Systems and Applications Series, Springer US, 2002, pp. 1–13. doi:10.1007/978-1-4615-1497-8_1.

[6] G. Wang, T. Wong, C. Yu, A computational model for multi-agent e-commerce negotiations with adaptive negotiation behaviors, Journal of Computational Science 4 (3) (2013) 135–143.

[7] A. J. Trappey, C. V. Trappey, W.-C. Ni, A multi-agent collaborative maintenance platform applying game theory negotiation strategies, Journal of Intelligent Manufacturing 24 (3) (2013) 613–623.

[8] W. Can, L. Yong-Quan, Research on a novel multi-agent negotiation model for e-commerce platform, in: Intelligent Computation Technology and Automation (ICICTA), 2014 7th International Conference on, IEEE, 2014, pp. 492–495.

[9] B. Y. Moeliodihardjo, B. W. Soemardi, S. S. Brodjonegoro, S. Hatakenaka, University, industry, and government partnership: Its present and future challenges in indonesia, Procedia - Social and Behavioral Sciences 52 (2012) 307 – 316, 10th Triple Helix Conference 2012. doi:http://dx.doi.org/10.1016/j.sbspro.2012.09.468.
URL http://www.sciencedirect.com/science/article/pii/S1877042812039237

[10] M. Pty. Ltd, Moodle registered sites on country based. (2016) [cited 08 March 2016].
URL https://moodle.net/sites/index.php?country=ID

[11] K. Brandl, Are you ready to "moodle", Language Learning & Technology 9 (2) (2005) 16–23.

[12] T. Saaty, Fundamentals of Decision Making and Prority Theory with the Analytic Hierarchy Process, Analytic hierarchy process series, Rws Publications, 1994.

[13] R. Obe, L. Hsu, PostgreSQL: Up and Running, O'Reilly Media, Inc., 2012.