Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 127–136
Marseille, 20-25 June 2022

User Interest Modelling in Argumentative Dialogue Systems

Annalena Aicher[1], Nadine Gerstenlauer[1], Wolfgang Minker[1], Stefan Ultes[2]
[1]Ulm University, Albert-Einstein-Allee 43, 89075 Ulm, Germany
[2]Mercedes-Benz Research and Development, Sindelfingen, Germany
annalena.aicher@uni-ulm,de

Abstract

Most systems helping to provide structured information and support opinion building, discuss with users without considering their individual interest. The scarce existing research on user interest in dialogue systems depends on explicit user feedback. Such systems require user responses that are not content-related and thus, tend to disturb the dialogue flow. In this paper, we present a novel model for implicitly estimating user interest during argumentative dialogues based on semantically clustered data. Therefore, an online user study was conducted to acquire training data which was used to train a binary neural network classifier in order to predict whether or not users are still interested in the content of the ongoing dialogue. We achieved a classification accuracy of 74.9% and furthermore investigated with different Artificial Neural Networks (ANN) which new argument would fit the user interest best.

Keywords: Interest Model, User Model, Argumentative Dialogue Systems (ADS), Conversational Systems, User Usability/Satisfaction, HCI, Preference Modelling

1. Introduction

In the last two decades due to the rapid development of the internet the amount of available information has increased rapidly. This flood of information has become very non-transparent and often even contradictory, making it difficult for humans to find the information they are looking for. Therefore, personalized systems address this overload by building and representing information customized for individual users (Gauch et al., 2007). According to Lu et al. (2015) personalized recommendation is the most efficient and promising solution to information overload, e.g. in e-commerce and online social platforms (Gogna and Majumdar, 2015). Such customization may include filtering for relevant information and/or identifying information of likely interest for the user (Gauch et al., 2007). As Mairesse and Walker (2010) pointed out a number of studies strongly suggest that dialogue systems that adapt to the user are more effective. Especially, regarding argumentative dialogues on controversial topics it is important to attract the user's interest to ensure an ongoing motivation to continue the conversation. The interest of a single user can be measured using explicit or implicit user feedback. In general, systems collecting implicit information place little or no burden on the user and thus, are more likely to be used (Gauch et al., 2007). Hence, in this paper we introduce a novel model to implicitly estimate the user interest during an argumentative dialogue. In analogy to user interest during web browsing discussed in (Yi et al., 2017), we assume that users which tend to have a long conversation, presumably are interested in the topic. This model is incorporated in the argumentative dialogue system BEA ('Building Engaging Argumentation') which we introduced in previous work (Aicher et al., 2021). The purpose of BEA is to engage in a deliberative dialogue with a human user in order to support their opinion building process by incrementally presenting automatically extracted arguments. As we pursue a cooperative exchange of arguments without trying to persuade the user, it is important which arguments are of particular interest to the user and thus, keep them motivated to continue the interaction. Therefore, BEA calculates after each user utterance the current interest on different aspects (so-called "meta-cluster") based on our interest model. Using a neural network classifier, the system detects when the user loses interest in the currently discussed cluster. To acquire the required training data we conducted a crowd-sourcing study with 292 participants. This classifier is integrated in BEA in order to adapt the dialogue strategy accordingly and thus, keep the conversation going. Both machine learning and rule-based approaches are discussed as potential adaption strategies.

The remainder of this paper is structured as follows. Section 2 gives an overview over related work. A description of the relevant components of BEA are given in Section 3. Our proposed user interest model is introduced in Section 4 which is followed by a description of its experimental usage, i.e. data acquisition for the ANN interest classification in Section 5. Subsequently, the results are discussed in Section 6. We close with a conclusion and a brief discussion of future work in Section 7.

2. Related Work

To provide personalized applications that fit user needs, it is common to build user profiles e.g. from heterogeneous information associated with an individual user or a group of users showing similar interests (Das et al., 2012; Gauch et al., 2007; Su et al., 2012). In order to measure the interest of a single user, existing research distinguishes between explicit and implicit user feedback (Gauch et al., 2007). For example, Amazon.com uses customer history records to recommend books, and the movie streaming provider Netflix recommends

movies to different users according to their individual browsing and watching records, as well as explicit ratings of seen movies/TV shows (Ricci et al., 2011; Hawashin et al., 2019). In general, there exist two types of filtering mechanisms based on explicit user ratings regarding recommender systems. First, collaborative filters (Pavlov and Pennock, 2002; Chien and George, 1999; Gazdar and Hidri, 2020) which use the user-user similarity principle stating that if a user highly rated an item, similar users would probably highly rate that item. Second, content-based filters (Pazzani and Billsus, 2007; Son and Kim, 2017) which recommend items based on the item-item similarity principle stating that if users highly rated an item, they would highly rate similar items. Hawashin et al. (2019) point out that explicit user rates do not always indicate the true hidden interest of the user. Henceforth they suggest to take not only explicit user rates and the time factor into account but also the user actual interests Hawashin et al. (2019).

We circumvent these difficulties by using implicit user feedback. According to Gauch et al. (2007) implicit feedback places less burden on the user, and as it automatically updates during the user-system-interaction. As implicit methods perform as well or better in practice than those collecting explicit feedback, they conclude it is the preferable approach. Still, the implicit detection of interest from user behaviour is more complicated and only scarcely researched. Especially, in the context of Argumentative Dialogue Systems (ADS) foremost explicit feedback channels are used. For example, Rach et al. (2020) asked users to evaluate dialogue content during an ongoing conversation [1]. However, explicit user feedback (Su et al., 2012) requires user responses that are not content-related and disturb the dialogue flow. Due to the described drawbacks of explicit methods and to achieve our aim of a natural interaction we chose to assess user interest implicitly.

Most approaches in human-machine argumentation focuses on competitive scenarios (Slonim et al., 2021; Rosenfeld and Kraus, 2016; Rakshit et al., 2017; Le et al., 2018) aiming to win a debate against, persuade or convince the user. Therefore, the modeling, assessment of and adaption to user interest is not considered in existing literature. Hadoux and Hunter (2021) modeled a framework that incorporates the beliefs and concerns of an opponent and Chalaguine and Hunter (2020) used a previously crowd-sourced argument graph and considered concerns of the user to persuade them. In contrast we pursue a cooperative exchange of arguments to help the user to a build well-founded opinion on a controversial topic (Aicher et al., 2021). To ensure the user's motivation to keep up the conversation it is important to take the user's current interests into account. Likewise to the approach of Zeng et al. (2020) who captures the temporal aspects of user interests in online conversation recommendation, our proposed interest model is updated continuously to assure that current and evolving user interest is captured.

To the best of our knowledge, existing literature lacks any reference to interest modelling in (argumentative) dialogue systems. Therefore, we developed a novel user interest model for argumentative dialogue systems based upon some parallels between user interest during web browsing and ADS. Commonly used in e-commerce (Chalyi and Pribylnova, 2019), most research projects deal with user interest in website content-based on browsing history (e.g. (Qiu and Cho, 2006; Zhou et al., 2019)). Yi et al. (2017) describe user interest by the user's browsing time and the content of webpages. They claim that rather short website content and long browsing time correlate with great user interest. Thus in general, one can say that tending to have a long conversation, a person may be interested in the topic at hand. Based on this idea, we developed our user interest model for ADS which is elaborated in Section 4. Furthermore, we are the first to incorporate also the loss of interest in our model as according to Gauch et al. (2007) implicit feedback techniques in general lack the ability to capture negative feedback.

3. The Argumentative Dialogue System BEA

In the following the relevant aspects of our argumentative dialogue system "Building Engaging Argumentation" (BEA) are introduced. After an overview on its knowledge base, the argument clustering we need for our interest model as well as the underlying dialogue model and interface of BEA are described. For further reference and a more detailed explanation of the whole framework we refer to our previous work (Aicher et al., 2021).

3.1. Knowledge Base

The herein utilized annotation scheme was introduced for annotating argumentative discourse structures and relations in persuasive essays by Stab and Gurevych (2014). They structure arguments in several components (*major claim*, *claim* and *premise*). The overall topic of the debate is called *major claim* representing the root node in the graph. *Claims* are assertions which formulate a certain opinion targeting the *major claim* but still need to be justified by further arguments, *premises* respectively. As shown in Figure 1 we consider two relations between these argument components (nodes), *support* (green arrow) or *attack* (red arrow). We choose a non-cyclic tree structure, where each node ("parent") is supported or attacked by its "children". If no children exist, the node is a leaf and marks the end of a branch. According to Wilcock and Jokinen (2021) in scenarios that do not adhere to a clear structure regarding speaking time and turn taking (like debates), extensive utterances presented by synthetic speech are hard to follow and understand. To prevent

[1] Rating categories: interesting, convincing, comprehensible and related

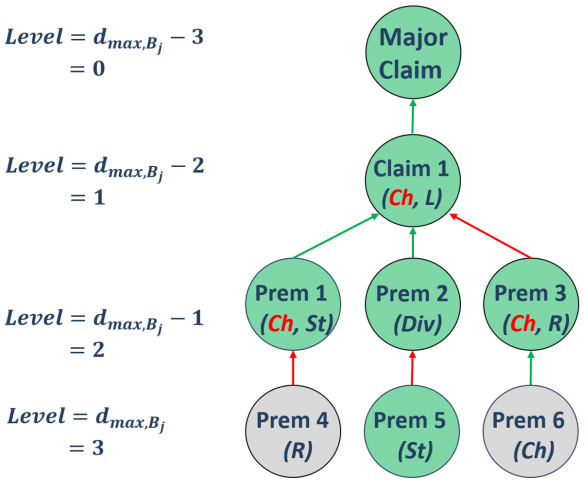

Figure 1: Visualisation of an exemplary argumentation tree structure with different argument components (visited nodes marked in green, unvisited ones in grey). The according associated clusters are denoted in brackets *Ch(ildren)* is marked in red due to an example calculation given in 4.1.3.

the user from being overwhelmed by the amount of information, in contrast to our previous work (Aicher et al., 2021) we introduce the available arguments incrementally depending on the user's request. The sample debate on the topic *Marriage is an outdated institution* which was thoroughly discussed by Rach et al. (2018) provides a suiting argument. It serves as knowledge base for the arguments and is taken from the *Debatabase* of the idebate.org[2] website. It consists of a total of 72 argument components (1 *major claim*, 10 *claims* and 61 *premises*) and their corresponding relations and encoded in an OWL ontology (Bechhofer, 2009) for further use. In each "why pro/con" move a single argument component is presented to the user. The maximal depth of a branch d_{max,B_j} varies from 5 up to 10. Due to the generality of the annotation scheme, the system is not restricted to the herein considered data. In general, every argument structure that can be mapped into the applied scheme can be processed by BEA.

3.2. Argument Clustering

Our user interest model requires semantically clustered arguments, such that each argument belongs to one or more meta-aspects (clusters) of the discussed topic. There are many different approaches for clustering data. Research in argument clustering is mostly based on textual structures or linguistic features using agglomerative clustering (Boltužić and Šnajder, 2015; Rakshit et al., 2019). However, as an argument can address more than one aspect of a topic, it may belong to multiple overlapping clusters (Daxenberger et al., 2020). Thus, according to Reimers et al. (2019), simple partitioning algorithms such as agglomerative clustering are not suitable for argument clustering. As machine learning techniques to identify semantic clusters are very complex and exceed the scope of this paper, we chose to apply manual clustering by three human expert annotators and will focus on the former in future work. Due to the fact that manual clustering captures semantically fine-grained nuances it may even be better in estimating the similarity of arguments (Daxenberger et al., 2020). Considering our sample dataset ten different clusters were identified: *Alternative relationships and parenthoods, Children, Divorce, Expectations and commitment, Harmful relationships, Law, Relationship stability, Religion, Remarriage, Social Acceptance*. Each argument directly addresses to one or more clusters. As each argument component targets the predecessor above it, it refers indirectly to all predecessing parents. Therefore, we define that each argument component inherits the clusters of its preceding nodes, i.e. it indirectly addresses all clusters its parent directly or indirectly addresses. Note that an argument component can both directly and indirectly address the same cluster, i.e. if it belongs to a cluster itself and it also inherits the cluster from its parent. The major claim denoting the overall topic does not belong to a cluster.

3.3. Dialogue Model

The interaction between the system and the user is separated in turns, consisting of a user action and corresponding natural language answer of the system. Table 1 shows the relevant[3] possible actions (moves) the user is able to choose from. The user can navigate through the argument tree and enquire more information. The determiners show which moves are available depending on the position of the current argument (root / parent node / "leaf" node). After the system's greeting the resulting dialogue is determined only by the user and their choices. The system response is based on the original textual representation of the argument components, which is embedded in moderating utterances. To support the impression of a natural conversation and to engage the user, personalized system responses are used, e.g. "An argument in favor is that.../Let us get back to the argument stating that...".

3.4. Interface

The graphical user interface (GUI) of BEA is illustrated in Figure 2. It either provides a drop-down menu or speech input. To detect possible differences between both modalities, we conducted our user study with two groups for each modality (see Section 5). In

[2] `https://idebate.org/debatabase` (last accessed 23th July 2020). Material reproduced from `www.iedebate.org` with the permission of the International Debating Education Association. Copyright © 2005 International Debate Education Association. All Rights Reserved.

[3] The whole set of available moves is presented in (Aicher et al., 2021). In the following only the moves which are relevant for the user interest model will be discussed.

Move	Description	Determiners
Level up	Switches to parent argument	Always (except for root)
Exit	Terminates conversation	Always
Why$_{pro}$	Information-seeking for a pro argument	If supporting child node exists
Why$_{con}$	Information-seeking for a con argument	If attacking child node exists

Table 1: Description of relevant moves the user can choose from in each turn of the interaction.

Figure 2: GUI of the baseline system with folded drop-down menu. Above the drop-down menu the dialogue history is shown. On the left side the sub-graph of the current branch is visible.

the drop-down system users can choose their action by clicking, whereas in the speech system a NLU framework introduced by Abro et al. (2021) processes the spoken user utterance. This input is captured with a browser-based audio recording that is further processed by the `Python` library `SpeechRecognition` using Google Speech Recognition. Its intent classifier uses the BERT Transformer Encoder presented by Devlin et al. (2018) and a bidirectional LSTM classifier. The system-specific intents are trained with a set of sample utterances of a previous user study. After a user intent is recognized, the spoken system response is presented using the Speech Synthesis of Web Speech API.
Regarding the findings of Yi et al. (2017) for user interest during web browsing (see Section 4) users are always aware of the length of the current page and their current position by looking at the page's scrollbar. To be able to draw an analogy to our model, a graphical visualization shows the users the length and structure of the currently visited argument branch. As shown in Figure 2 left to the dialogue history, the current root of the argument branch, the argument branch itself and the user's current position (green bordered node) are displayed. Already visited nodes are shown in green and unknown ones in blue. For creating the graphs, we use the `Python` package `pydot`.

4. User Interest Model

To develop a flexible system that takes into account the individual user interests, one has to resolve the issue of how to estimate the latter. As mentioned in Section 2, state-of-the-art approaches require explicit user feedback. We follow an implicit approach to ensure a natural, content-based dialogue. To the best of our knowledge, all of the existing implicit approaches deal with the modelling of user interest in website content, based on browsing history and reading time, without reference to dialogue systems. Thus, we build our user interest model upon some parallels we identified in the work of Yi et al. (2017). They state that rather short website content and long browsing time correlate with a bigger user interest. Likewise, the main idea incorporated in our model states that having a longer conversation, a person has a greater interest in the currently discussed topic. Most importantly length here does not refer to temporal length due to the fact that BEA's responses, i.e. the presented argument components, vary in their length, which the user cannot know beforehand. Furthermore, some available moves (e.g. general information on the interaction) are not content-related and thus, should not be included in the interest model. Therefore, we focus on the number of requested arguments on a certain cluster in relation to the number of all available arguments. In analogy to Yi et al. (2017), we consider content-dependent areas (branches) of our argument tree structure. As described in Subsection 3.2, each node inherits the clusters of its predecessors due the logical structuring where each argument component substantiates the previous one. Thus, the length of the visited argument branch is taken into account. Each component (except for "leaf nodes") directly addressing a cluster represents the root node of a relevant subtree, that is considered in the calculation. According to their definition subtrees which are assigned to the same cluster may overlap and thus, count multiple times, in particular when an argument component directly addresses a cluster that one of its predecessors has already addressed (see Figure 1 for the cluster *Ch(ildren)*. Hence, it is taken into account that the user explicitly requests for further information on an already introduced cluster and thus, shows an increased interest in the latter.
In the following subsections we explain our user interest model consists and merge them at the end.

4.1. User Interest for each Subtree

Let B_{i_c} be the subtree with root node i_c[4] belonging to cluster c; $|B_{i_c}|$ the total number of descendants of i: If the user has visited $|B_{i_c,v}|$ of these descendants, the

[4]Where i denoted a unique identifier.

user interest with regard to subtree B_{i_c} is defined as

$$I\left(B_{i_c}\right) = \frac{|B_{i_c,v}|}{|B_{i_c}|} \tag{1}$$

with $I\left(B_{i_c}\right) \in [0,1]$[5].

4.1.1. Weight for Relative Subtree Size

Intuitively the overall class interest from the individual interest values $I\left(B_{i_c}\right)$ (see Equation (1)) can be determined by taking their average (Yi et al., 2017). Still, the average would consider all interest values equally without accounting for differences in the subtrees sizes. Thus, small subtrees would be over-represented and larger ones under-represented. To prevent this, we introduce a weight that accounts for different subtree sizes. To determine the overall cluster interest of c a subtree B_{i_c} is weighted with

$$\omega_{n,B_{i_c}} = \frac{|B_{i_c}|}{\sum_{k_c} |B_{k_c}|}, \tag{2}$$

with $\omega_{n,B_{i_c}} \in [0,1]$. $\omega_{n,B_{i_c}}$ displays the relative size of a subtree B_{i_c} by taking the number of arguments into account which are contained in all subtrees B_{k_c} with root nodes k belonging to cluster c.

4.1.2. Weight for Relative Subtree Size

Considering hierarchical argumentation structures, arguments on different levels differ in information detail. In particular, an argument component located at the beginning of a branch (smaller level) is more general than one deeper down (higher level). The latter provides in-depth information and contains much more details. Hence, we introduce a hierarchical weighting in order to incorporate the different levels of argument depth into the interest model by assuming that a user who is highly interested will ask for more detailed information. Therefore, subtrees starting at lower levels will be assigned larger weight values than subtrees with root nodes closer to the Major Claim. As the weights are required to have ascending values, the argument level is divided by the sum of levels of the current branch. Let node i be a descendant of the branch root node j. The maximum depth of the corresponding argument branch B_j is denoted by d_{max,B_j} and the level of i is denoted by d_{i,B_j}. We define the weight $\omega_{d,B_{i_c}}$ for subtree B_{i_c} with root node i belonging to cluster c:

$$\omega_{d,B_{i_c}} = \frac{d_{i,B_j}}{\sum_{l=1}^{d_{max,B_j}-1} l}, \tag{3}$$

with $\omega_d in [0,1]$. Note that for leaf nodes, no succeeding arguments exist and thus, no further information. Consequently, the upper boundary of the sum in the denominator of (4.1.2) ends one level above the leaf node as visualized in Figure 1.

[5] $I\left(B_{i_c}\right) = 0$ means that the user has not requested any further information on argument node i and consequently, is not interested in cluster c. $I\left(B_{i_c}\right) = 1$ means that all available information has been requested and thus, largest possible interest in c with regard to the presented subtree.

4.1.3. Overall User Interest Model

To determine the overall user interest on a specific cluster c, all subtrees of the argument structure tree which belong to c have to be considered. Thus, we iterate over all subtree root nodes k_c and add up all weighted subtree interest values. Thus, the overall user interest for cluster c is given by

$$I_c = \frac{\sum_{k_c} \omega_{d,B_{k_c}} \omega_{d,B_{k_c}} I\left(B_{i_c}\right)}{\sum_{k_c} \omega_{d,B_{k_c}} \omega_{d,B_{k_c}}}. \tag{4}$$

The interest values in Equation 4 are normalized, such that $I_c \in [0,1]$. Hence, $I_c = 1$ denotes the highest possible user interest in cluster c, whereas $I_c = 0$ denotes the exact opposite. For a better understanding an example calculation with respect to the argument tree shown in Figure 1 is shown in the following. In order to determine the user interest in the cluster "Children (Ch) three visited subtrees with root nodes $i_{Ch} \in Claim1, Prem1, Prem3$ have to be considered. As only children of $Claim1$ were visited, it follows that $I\left(B_{i_{Ch}}\right) = \frac{4}{6} = \frac{2}{3}$, whereas it is 0 for the other two subtrees. As there are three subtrees containing 6+1+1=8 nodes, it follows that $\omega_{n,B_{i_{Ch}}} = \frac{6}{8} = \frac{3}{4}$ for $Claim1$ and $\frac{1}{8}$ for the two others. By dividing the respective level (1 for $Claim1$ and 2 for the rest) by $d_{max,B_j} = 3$, we get $\omega_{d,B_{i_c}} = \frac{1}{3}$ and $\frac{2}{3}$. Merging these values according 4 it follows for the user interest with regard to the cluster *Ch* is:

$$I_{Ch} = \frac{\frac{2}{3}\frac{3}{4}\frac{1}{3} + 0 + 0}{\frac{3}{4}\frac{1}{3} + \frac{1}{8}\frac{2}{3} + \frac{1}{8}\frac{2}{3}} = 0.4,$$

which means that the user would seem moderately interested in this cluster.

5. Experimental Setup for Interest Classification

In order to recognize if the user loses interest in the current subtopic (cluster) and identify the next argument of greatest interest to the user we implemented two separate ANN classifiers. To acquire the required data a real user study was conducted which is described in the following followed by an overview over the applied data pre-processing and feature extraction.

5.1. Data Acquisition

To acquire the amount of data needed to train the user interest classifiers for interest loss and most interesting next argument, we conducted an online crowdsourcing study. The study was conducted online via the crowdsourcing platform "Crowdee" (`https://www.crowdee.com/`, 12-29th November 2021) with participants from the UK, US and Australia. All 292 participants were non-experts without a topic-specific background. After an introduction to the system (short text and demo video), all users got the same study task, which was to listen to enough arguments to build a well-founded opinion on the topic (at least ten). The

first 139 participants interacted with BEA via drop-down menu input, the other 153 via speech. The participants were paid 15.36€ per hour. The duration for the menu group was estimated at approx. 15 minutes and for the speech group at approx. 20 minutes. Before and after the interaction the participants had to answer questionnaires on their interaction with BEA. As these questionnaires concern the perception of the different modalities (menu vs. speech I/O), only relevant aspects (e.g. data exclusion criteria)) are presented in this paper.

During the dialogue the user interest was calculated as described in Section 4 after each user turn. After each move (except concerning moves addressing the major claim) the participants had to rate their interest on a 5-point Likert scale: 5 (extremely interesting), 4 (very interesting), 3 (moderately interesting), 2 (slightly interesting), 1 (not at all interesting).

The first claim [6] is presented after the user asks for a pro/con argument. It is the only claim which is chosen randomly as afterwards the participants have to choose from the remaining claims whenever they navigated back to the major claim[7]. After the user selects the most interesting claim and BEA continues the dialogue by presenting the chosen argument. Taking into account that uninterested users might want to quit the interaction, the participants were allowed to end the dialogue whenever they felt like having heard enough arguments (minimum: 10 arguments [8]) to build a well-founded opinion. In average the participants interacted with BEA for 30:41 minutes (menu: 26.26 min; speech: 34:32 min). This difference can be explained by the fact that the spoken interaction inherently takes longer than clicking on an option in a drop-down menu. Regarding both study groups the interactions were long enough to obtain sufficient data for the classifier training. Analyzing questionnaire answers and feedback we noticed that some participants seemed to have issues. Eleven datasets of users of the speech group showed inconsistencies. Further eight participants in the speech group and one in the menu group reported problems using BEA or stated that they did not understand the interaction. Hence, their data was excluded from our training dataset. This leads to a total number of data records of 272 participants (menu: 138, speech: 134).

5.2. Pre-processing and feature extraction

A central issue in machine learning is the selection of relevant features (Langley and others, 1994). For the herein presented classification tasks we chose several features based on the user interest model (see Section 4) and the user's preceding interaction with BEA.

[6]Due to the argument tree structure described in Subsection 3.1 the respective argument component is a root node of the corresponding branch.

[7]Using the "level up" user move

[8]The minimum amount of arguments was chosen to ensure that enough data could be collected.

In the following the features used for both classifiers and the necessary processing are shortly described.

5.2.1. Cluster Interest

The calculated current user interest on each of the ten cluster (based on Section 4) serves as an input feature. Since the interest values are in the range of 0 to 1 no further pre-processing is needed.

5.2.2. Number of Visited Arguments

The calculated user interest depends on the arguments the user has already listened to. If users are not interested in a certain cluster, they will not ask for further information on this cluster, resulting in I_c being near to or even equal to 0. Thus, as the interest values are initially instantiated with 0, can not be distinguished without the information of the number of visited arguments per cluster at each turn of the interaction. In order to normalize this respective input feature, the relative number of visited arguments in cluster c_i is determined by: $rc_{i,visited} = \frac{|c_{i,visited}|}{|c_{i,all}|} \in [0,1]$, where $|c_{i,visited|}$ denotes the number of visited arguments for cluster c_i and $|c_{i,all|}$ denotes the number of arguments belonging to cluster c_i.

5.2.3. Previous Move

A change in the user interest might depend on a previous move which influences the further course of the dialogue. Therefore, the previous user move encoded categorically and transformed into a respective binary representation is added to the feature vector.

5.2.4. Previous Claim

To determine the user's loss of interest it might be relevant which claim was heard before. Thus, the ten claims of our dataset are binary encoded analogous to the "previous move" and added to the feature vector.

5.2.5. Already Visited Claims

As it is relevant which claims have been visited and which have not (no claim should be presented twice) this information is incorporated in our feature vector as binary feature for each claim.

6. Results and Discussion

In the following the results of our classifiers are discussed, to 1) automatically recognize the loss of user interest in the current aspects of the topic and 2) find the best-fitting claim to continue after the former. Two separate ANN classifiers are implemented and several approaches are compared. Furthermore, a rule-based approach for predicting the next claim is shown.

6.1. Classification of User Interest Loss

We use an linear feedforward Artificial Neural Network implemented using `Tensor ow 2.0` and the integrated Keras API[9]. The network is trained 100 epochs

[9]https://www.tensor ow.org/api docs/python/tf/keras,(last accessed 2021-12-13)

using a training dataset consisting of 7814 samples (4621 menu, 3193 speech). As input features the calculated user interest values as well as the number of visited arguments per cluster were used. For the training three hidden layers (120 hidden neurons each) and the Rectified Linear Unit (ReLu) function as activation function were used.
To train a binary classifier, the participants' ratings on a 5-point Likert scale were divided in two groups: 4 ("very interesting") or 5 ("extremely interesting") is assigned to being interested; 3 ("moderately interesting") or lower as having lost interest. We used the binary cross-entropy loss function in the ANN and the sigmoid activation function in the output layer. To prevent overfitting and evaluate our classifier, 10-fold cross-validation was used. According to the results shown in Table 2 an overall accuracy[10] of 74,9% was obtained. The highest accuracy with 76,6% was reached for the speech group. As we aim for a general classifier independent of the I/O modalities, the classifier is still trained on all available data.

6.2. Next Claim Classification

Likewise to the previous subsection a feedforward ANN was used to classify which claim is most interesting to the users after they lost interest in the current cluster content. The training dataset consists of 1091 samples (menu: 564; speech: 527), where participant stated to be "moderately interested" or even less. Using this data, the network is trained 100 epochs. Whereas previously a binary classifier was used this classification task is trained on multiple classes. The classes we aim at predicting are the ten claims (argument components on the first level beneath targeting the major claim). The size of the input layer is equal to the number of features used for training. They are extracted and pre-processed as described in Subsection 5.2. For each approach described in the following we tried different combinations of input features. Since the calculated user interest and the number of visited arguments are integral, they are always used as input features. This basic feature vector is extended by the binary encoded previous move and the previous claim features. Furthermore, the information on already visited claims is added.

6.2.1. One-hot Encoded Multi-class Claim Classification

To predict the next claim an ANN is trained by using the possible claims as class labels. The output layer consisting of ten neurons is defined by one-hot encoding. The softmax function serves as activation and categorical cross-entropy as loss function. The results for the multi-class claim classifier are shown in Table 2. Using all input features and both modalities, we get the best accuracy with 28.5%. In contrast to the interest loss classifier in Section 6.1, separated consideration of both modalities leads to a lower accuracy. Varying the input features e.g. excluding the previous move from the feature vector, decreases the accuracy. Additionally excluding the previous claim features, the overall accuracy decreases to 24.5% (SD 1.8%). This can be explained by the fact, that the previous user action and claim have a direct influence the user's choice of a next claim. Thus, it is best to use all available features as well as both input modalities. Choosing out of ten claims the best fitting one, an accuracy of 28.5% for a classifier is noticeably better than random. However, we will investigate in the following if this result can be exceeded.

6.2.2. Binary Claim Classification of Cluster Groups

As claims can belong to more than one cluster, the multi-class multi-label classification to predict the clusters the next claim should belong to, yields in no result. Thus, we chose to identify claims belonging to the same clusters and those with overlapping clusters. Regarding our dataset four claims can clearly be distinguished and six claims show overlapping cluster affiliation, which were grouped accordingly. Upon this abinary ANN classifier with a single neuron in the output layer is defined, to predict to which group the next claim should belong. A binary cross-entropy loss function and the sigmoid function. Using the data of both modalities leads to an accuracy of 62.9%, as can be seen in Table 2[11]. Even though this result is better than random, considering a binary classification the accuracy is still rather low. This might be due to the fact, that the users were not explicitly asked to choose a cluster but the next claim which seems to has a significant influence.

6.2.3. Binary Classification of Claim Polarity

Whereas in the previous approaches we focused on argument clustering, in the following it is investigated if predictions about the polarity of the next claim can be made, i.e. whether the user would choose a supporting or attacking claim. We define two classes based on the argument polarity (five claims support and five claims attack the major claim). Thus, we train a binary classifier with a single neuron in the output layer that predicts the polarity of the next claim. Again a binary cross-entropy loss function and the sigmoid activation function in the output layer are used.
Even though we are able to detect a slight tendency with an accuracy of 62,9% (see Table 2) for both modalities, the prediction is not significantly better than a random guess. This might again be due to the fact, that the users were not asked to state which polarity they would choose next. Furthermore, the training features are tailored to the analysis of cluster interest. By including data that is concerned with the argument

[10] Accuracy for both systems: menu and speech.

[11] The best accuracy is achieved by excluding the input features of the previous move and claim.

Classification	**Accuracy**$_{both}$ in % (SD)	**Accuracy**$_{menu}$ (SD)	**Accuracy**$_{speech}$ (SD)
Interest	0.749 (0.023)	0.738 (0.017)	**0.766** (0.025)
Next Claim (Multi-Class)	**0.285** (0.039)	0.272 (0.071)	0.280 (SD 0.060)
Next Claim (binary Cluster)	0.629 (0.042)	**0.658** (0.088)	0.615 (0.073)
Next Claim (binary Polarity)	0.629 (0.035)	0.594 (0.072)	**0.643** (0.083)

Table 2: Accuracy of trained classifiers depending on the different modalities which are considered.

polarity, e.g. the current stance of the user, better results might be achieved, which will be an aspect of future work.

6.3. Rule-based Selection of the Next Claim

As the classification accuracy of the previous Subsections 6.2.1- 6.2.3 is not sufficient, we developed a rule-based approach based on our user interest model described in Section 4 to choose a suitable next claim. All unvisited claims are considered potential candidates. If there is only one claim left, this one will be presented to the user. Otherwise, a claim belonging to the cluster with the highest interest value is chosen. However, more than one unvisited claim may belong to the respective cluster. Therefore, a claim which belongs to clusters with high user interest and does not belong to uninteresting clusters is chosen. Therefore, for each eligible claim the average interest of all the clusters it belongs to is determined and the highest value is selected. If there are still multiple claims remaining, a random one is chosen and presented to the user.

7. Conclusion and Future Work

In this work, we introduced a novel user interest model for estimating user interest during argumentative dialogues. We used ANN classifiers in order to detect dynamical changes in user interest. Based upon this, we enable our argumentative dialogue system BEA to proactively intervene when users seem to loose interest and keep up the conversation by suggesting arguments that meet their interest.

To the best of our knowledge, our user interest model is the first model for implicitly estimating the user interest during an ongoing argumentative dialogue. State-of-the-art approaches mostly cover explicit feedback mechanisms and do not consider cooperative, argumentative scenarios. Our presented model considers various factors like the length of the dialogue, the number of arguments the user listened to and the level of detail of the visited. As this user interest model is not tailored to our sample dataset, it can be used in every argumentative dialogue system with a tree-like data structure. However, the arguments need a semantic clustering. As manual semantic clustering is not scalable for broader application especially considering larger datasets, machine-aided clustering is inevitable. Therefore, it will be subject to future work to find a suitable method to cluster arguments automatically.

In order to be able to apply machine learning we conducted an online user study for data acquisition. Participants were asked to use the system while constantly giving feedback on their current interest level and the claims they were most interested in. Furthermore, we estimated their interest during the study based on our user interest model. Using the acquired data we trained a binary neural network classifier in order to predict whether or not users are still interested in the content of the ongoing dialogue. We achieved a classification accuracy of 74.9%. Moreover, it was investigated how the system can predict which claim might be best presented to the user. Different approaches on ANN multi-class classification were explored but none of them showed reliable results. This can be explained by the fact that each claim can belong to multiple clusters and thus, containing similar, indistinguishable content. Furthermore, if a user is interested in multiple clusters to a similar degree, it is unclear which of those clusters should be presented. Since multi-class classification did not yield satisfying results, we developed a rule-based approach which presents the claim with the highest cluster interest. In case users refuse the suggestion they are able to switch back to the previous argument. Hence frustration due to unexpected system behaviour is minimized and the user's trust in the system is strengthened (Kraus et al., 2020).

In future work we will evaluate the herein described extension of BEA in an extended user study. As currently BEA provides a randomly chosen argument when the user requests further information, we want to investigate the influence if this choice is based on our user interest model. Last but not least we aim to compare and combine our approach with the prevention of confirmation bias and therefore, providing unbiased and balanced information which is still interesting to the user and thus, keeping the user motivated to interact with BEA.

8. Acknowledgements

This work has been funded by the DFG within the project "BEA - Building Engaging Argumentation", Grant no. 313723125, as part of the Priority Program "Robust Argumentation Machines (RATIO)" (SPP-1999).

9. Bibliographical References

Abro, W. A., Aicher, A., Rach, N., Ultes, S., Minker, W., and Qi, G. (2021). Natural language understanding for argumentative dialogue systems in the opinion building domain. *arXiv*, arXiv:2103.02691.

Aicher, A., Rach, N., Minker, W., and Ultes, S. (2021). Opinion building based on the argumentative dialogue system bea. In *Increasing Naturalness and Flexibility in Spoken Dialogue Interaction: 10th International Workshop on Spoken Dialogue Systems*, pages 307–318. Springer Singapore.

Bechhofer, S. (2009). Owl: Web ontology language. In *Encyclopedia of Database Systems*, pages 2008–2009. Springer.

Boltužić, F. and Šnajder, J. (2015). Identifying prominent arguments in online debates using semantic textual similarity. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 110–115, Denver, CO, June. Association for Computational Linguistics.

Chalaguine, L. A. and Hunter, A. (2020). A persuasive chatbot using a crowd-sourced argument graph and concerns. In *COMMA*.

Chalyi, S. and Pribylnova, I. (2019). The method of constructing recommendations online on the temporal dynamics of user interests using multilayer graph. *EUREKA: Physics and Engineering*, (3):13–19.

Chien, Y.-H. and George, E. I. (1999). A bayesian model for collaborative filtering. In *AISTATS*.

Das, R., Farrell, R. G., and Rajput, N. (2012). Social recommender system for generating dialogues based on similar prior dialogues from a group of users, September 25. US Patent 8,275,384.

Daxenberger, J., Schiller, B., Stahlhut, C., Kaiser, E., and Gurevych, I. (2020). Argumentext: argument classification and clustering in a generalized search scenario. *Datenbank-Spektrum*, 20(2):115–121.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Gauch, S., Speretta, M., Chandramouli, A., and Micarelli, A. (2007). User profiles for personalized information access. *The adaptive web*, pages 54–89.

Gazdar, A. and Hidri, L. (2020). A new similarity measure for collaborative filtering based recommender systems. *Knowledge-Based Systems*, 188:105058.

Gogna, A. and Majumdar, A. (2015). A comprehensive recommender system model: Improving accuracy for both warm and cold start users. *IEEE Access*, 3:2803–2813.

Hadoux, E. and Hunter, Anthony, e. a. (2021). Strategic argumentation dialogues for persuasion: Framework and experiments based on modelling the beliefs and concerns of the persuadee. In *arXiv*, volume 2101.11870.

Hawashin, B., Aqel, D., AlZu'bi, S., and Jararweh, Y. (2019). Novel weighted interest similarity measurement for recommender systems using rating timestamp. In *2019 Sixth International Conference on Software Defined Systems (SDS)*, pages 166–170. IEEE.

Kraus, M., Wagner, N., and Minker, W. (2020). Effects of proactive dialogue strategies on human-computer trust. In *Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization*, pages 107–116.

Langley, P. et al. (1994). Selection of relevant features in machine learning. In *Proceedings of the AAAI Fall symposium on relevance*, volume 184, pages 245–271.

Le, D. T., Nguyen, C.-T., and Nguyen, K. A. (2018). Dave the debater: a retrieval-based and generative argumentative dialogue agent. *Proceedings of the 5th Workshop on Argument Mining*, pages 121–130.

Lu, J., Wu, D., Mao, M., Wang, W., and Zhang, G. (2015). Recommender system application developments: A survey. *Decision Support Systems*, 74:12–32.

Mairesse, F. and Walker, M. A. (2010). Towards personality-based user adaptation: psychologically informed stylistic language generation. *User Modeling and User-Adapted Interaction*, 20(3):227–278.

Pavlov, D. and Pennock, D. M. (2002). A maximum entropy approach to collaborative filtering in dynamic, sparse, high-dimensional domains. In *NIPS*, volume 2, pages 1441–1448. Citeseer.

Pazzani, M. J. and Billsus, D. (2007). Content-based recommendation systems. In *The adaptive web*, pages 325–341. Springer.

Qiu, F. and Cho, J. (2006). Automatic identification of user interest for personalized search. In *Proceedings of the 15th international conference on World Wide Web*, pages 727–736.

Rach, N., Langhammer, S., Minker, W., and Ultes, S. (2018). Utilizing argument mining techniques for argumentative dialogue systems. In *Proceedings of the 9th International Workshop On Spoken Dialogue Systems (IWSDS)*, May.

Rach, N., Matsuda, Y., Daxenberger, J., Ultes, S., Yasumoto, K., and Minker, W. (2020). Evaluation of argument search approaches in the context of argumentative dialogue systems. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 513–522, Marseille, France, May. European Language Resources Association.

Rakshit, G., Bowden, K. K., Reed, L., Misra, A., and Walker, M. A. (2017). Debbie, the debate bot of the future. In *Advanced Social Interaction with Agents - 8th International Workshop on Spoken Dialog Systems*, pages 45–52.

Rakshit, G., Bowden, K. K., Reed, L., Misra, A., and Walker, M. (2019). Debbie, the debate bot of the

future. In *Advanced Social Interaction with Agents*, pages 45–52. Springer.

Reimers, N., Schiller, B., Beck, T., Daxenberger, J., Stab, C., and Gurevych, I. (2019). Classification and clustering of arguments with contextualized word embeddings. *arXiv preprint arXiv:1906.09821*.

Ricci, F., Rokach, L., and Shapira, B., (2011). *Introduction to Recommender Systems Handbook*, pages 1–35. Springer US, Boston, MA.

Rosenfeld, A. and Kraus, S. (2016). Strategical argumentative agent for human persuasion. In *ECAI'16*, pages 320–328.

Slonim, N., Bilu, Y., Alzate, C., Bar-Haim, R., Bogin, B., Bonin, F., Choshen, L., Cohen-Karlik, E., Dankin, L., Edelstein, L., et al. (2021). An autonomous debating system. *Nature*, 591(7850):379–384.

Son, J. and Kim, S. B. (2017). Content-based filtering for recommendation systems using multiattribute networks. *Expert Systems with Applications*, 89:404–412.

Stab, C. and Gurevych, I. (2014). Annotating argument components and relations in persuasive essays. In *COLING*, pages 1501–1510.

Su, Z., Yan, J., Ling, H., and Chen, H. (2012). Research on personalized recommendation algorithm based on ontological user interest model. *Journal of Computational Information Systems*, 8(1):169–181.

Wilcock, G. and Jokinen, K. (2021). Towards increasing naturalness and flexibility in human-robot dialogue systems. In *Increasing Naturalness and Flexibility in Spoken Dialogue Interaction: 10th International Workshop on Spoken Dialogue Systems*, pages 109–114. Springer Singapore.

Yi, J., Zhang, Y., Yin, M., and Zhao, X. (2017). A novel user-interest model based on mixed measure. In *Journal of Physics: Conference Series*, volume 887, page 012061. IOP Publishing.

Zeng, X., Li, J., Wang, L., Mao, Z., and Wong, K.-F. (2020). Dynamic online conversation recommendation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3331–3341, Online, July. Association for Computational Linguistics.

Zhou, G., Mou, N., Fan, Y., Pi, Q., Bian, W., Zhou, C., Zhu, X., and Gai, K. (2019). Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 5941–5948.