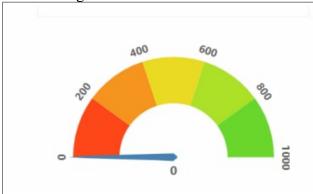
CustomerAgreementBarometer



0-200: Confident refusal to buy

200-400: Unwillingness to make a purchase, doubts with a preference to refuse a purchase, a complaint about the terms of purchase

400-600: Doubts about the purchase

600-800: Doubts with a preference to make a

purchase

800-1000: Confident desire to make a purchase

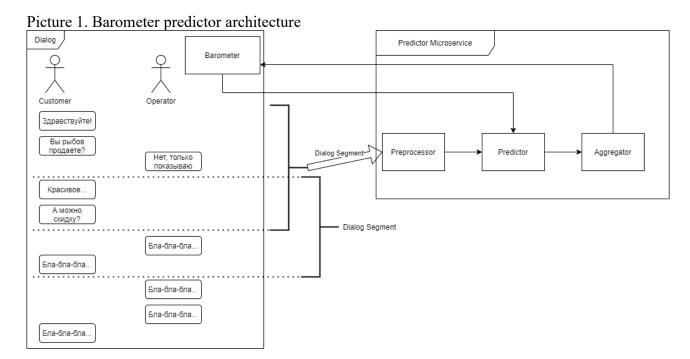
The microservice is developed to predict the customer agreement barometer based on a dialog between a customer and seller.

The input data is previous barometer values and a dialog segment. The output data is a new barometer value. Previous and predicted batometer values are normalized to the [0,1].

Each dialog segment contains the following parts:

- 1. The user's messages combined from the beginning of the dialog or from the last message of the seller to the next message of the seller.
- 2. The seller's response to the user's messages, messages combined from the user's messages from paragraph 1 to the next user's messages.
- 3. The user's reaction to the seller's response, messages combined from the seller's response from paragraph 2 to the end of the dialog or to the next message from the seller.

This segmentation of the dialog, combined with the use of the previous barometer value, allows to save the context of the dialog during the evaluation process.



The model is trained using cloud services and saved to the file. Predictor is launched as python app that loads model from file. Predictor is launched in the Docker container.

The following techniques are used during training and running:

- 1. Lemmatizing
- 2. TF-IDF (term frequency inverted document frequency)
- 3. RandomForestRegressor

Picture 2. Training schema Prepare a common vocabulary Training data ∏ InitCustomerMessage FollowingCustomerMessage Lemmatizer $Find\ optimal\ training\ meta-parameters,\\ see \ https://en.wikipedia.org/wiki/Hyperparameter_optimization\#Grid_search\\ and\ https://en.wikipedia.org/wiki/Cross-validation_(statistics)$ Words that are not represented in the vocabulary are ignored TF-IDF Group together the inflected forms of a word, see https://en.wikipedia.org/wiki/Lemmatisation Vocabulary is common, but the features calculated Vocabulary independently for each component of dialog segment 2. GridSearchCV Sparse --' matrices for messages Lemmatized Messages InitCustomerMessage TF-IDF Lemmatizer FollowingCustomerMessage Result RandomForestRegressor PreviousBarometer,

Transform text to sparse matrices that are more convenient for ML, see https://en.wikipedia.org/wiki/Tf-idf

For the TF-IDF words and bigrams are used

Value from [0,1]

Normalized to [0,1]