Semantic Data Fragmentation for Identification of Covariant Conceptual Drift in Machine Learning Models

I.Yu. Kashirin

***Abstract*— Such types of data drift in ML models as actual positive and negative drifts, as well as varieties of fragmentary drifts in different directions, are considered. An overview of the current state of the problem is given, which highlights the methods of sliding window, trigger ensemble of models, covariant shift, personalization drift correction, season correction, online learning method, low precision sampling, monitoring and clipping features. A modernized theory of binary relations was used to design the new method. As an example, the subject area "communication services" is considered, for which a special architecture of the ontological knowledge model is designed. The new drift correction method is the basis of a new technology for designing classification, regression and forecasting models for specifically formalized subject areas. When choosing the scope of the "sliding window", the structure of the knowledge model is taken into account first of all. The input features of the training data set are grouped according to the structure of the concepts and relationships of the knowledge base. The resulting data drift compensation technology makes it possible to improve the ROC AUC accuracy characteristics of ML models from 0.74 to 0.80, which makes it possible to evaluate the technology as an effective means of automatic correction. The new drift correction method is the basis of a new technology for designing classification, regression and forecasting models for specifically formalized subject areas. When choosing the scope of the "sliding window", the structure of the knowledge model is taken into account first of all. The input features of the training data set are grouped according to the structure of the concepts and relationships of the knowledge base.**

***Keywords*— Conceptual drift, machine learning, positive and negative prediction .**

I. Introduction

The problem of data drift and concepts in machine learning models [1] arises over time due to a decrease in the accuracy of forecasting or classification [2]. In this case, the model trained on earlier data does not provide its functions on later input datasets.

We list the reasons for the change in the ratios in the current data in comparison with the previous data:

- changing the sources of the input data set;
- demographic changes over the years;
- emergence of new high technologies;
- identification of previously unexplored factors of influence in the analyzed subject area;
- influence of periodic natural phenomena;
- shifting the focus of user interests to new topics.

To overcome the drift of concepts in the analyzed subject area, it is necessary to carefully implement all stages of the life cycle of the development and operation of machine learning models.

In the most famous Data Minig design approach (CRISP-DM [3]), the life cycle was planned as a sequence of such activities as: domain understanding, data understanding, data preparation, model creation, evaluation and implementation.

With the severity of the concept drift problem identified, it makes sense to represent the life cycle with a new sequence (Fig. 1). In Figure 1, the solid line indicates the sequence of stages in the design and operation of models adapted to concept drift. The dotted line corresponds to the use of automated tools for monitoring, re-estimating and online model correction. It follows from the figure that data monitoring is required at each of the stages of designing and operating predictive models.

II. Life Cycle of Machine Learning Model

This article publishes a new method for identifying the problem of concept drift, called the semantic fragmentation method (SFM). To present the essence of the SFM, a formal apparatus will be used, using the concept of probability distribution [4].

SFM is used in binary classification problems where two subclasses of target labels are interpreted as "positive" and "negative" in prediction, corresponding to the desired outcome and the undesirable outcome from the point of view of the decision maker. The same can be attributed to the vectors of input features, the change of which during covariant drift can be divided into "positive shift", "neutral shift" and "negative shift".

III. Formal description of the basic drift concept

If you are using *Word*, use either the Microsoft Equation Editor or the *MathType* add-on (http://www.mathtype.com) for equations in your paper (Insert | Object | Create New | Microsoft Equation *or* MathType Equation). “Float over text” should *not* be selected.

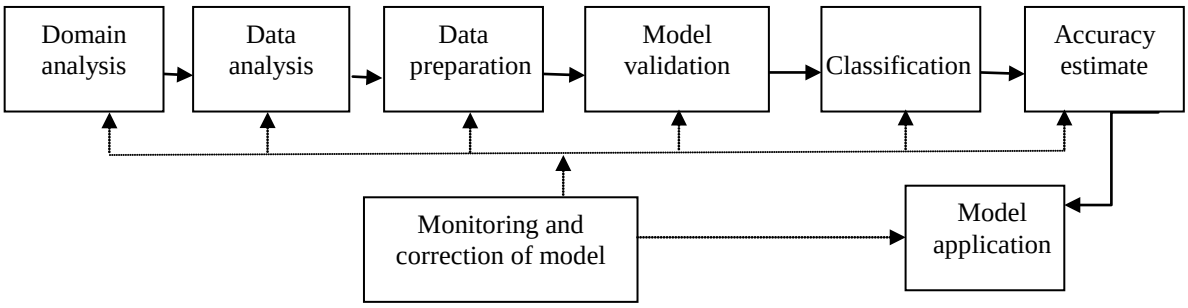

FIGURE 1. Life cycle of machine learning model

IV. FORMAL DESCRIPTION OF THE BASIC DRIFT CONCEPT

The following is a formal description of the basic concepts on which the SFM is based, and a brief listing of the existing methods for detecting concept drift.

Let *P(X)* be the unconditional probability of obtaining the feature vector *X* in the input data set, and *P(y|X)* the conditional probability of obtaining the values of the target variable *y* given the input features *X*.

In the study of data drift relative to a predictive or classifying ML model, two main types of drift are distinguished: *drift of feature instances* [5-10] and *drift of the concept system* [11-18].

Drift of feature instances (data drift) implies a change in *P(X)* caused by a change in the input data set. The dependence of the feature vector *X* on the target variable y remains unchanged. In this case, a change in *P(X)* may entail a change in *P(y|X)*. In order to identify data drift, it is first necessary to identify a change in the data distribution of the vector *X*.

For example, the drift of attribute instances can be considered as a change in the values of the attribute "quantitative decrease in monthly purchases" or the values of the attribute "number of site visits in the last 24 hours".

Drift of the system of concepts (drift of the concept) is a change *in P(y|X)*, i.e. change in the dependence of the target variable y on the vector of features *X*. The drift of the concept system suggests that the distribution of features itself can be unchanged, however, the patterns revealed by the ML model begin to be revealed less reliably. As a result, trained models become inefficient. The initial exploration of concept drift involves identifying changes in the distribution of the target variable y and/or changing the dependence of *y* on *X*. The basic relation that identifies the drift of a system of concepts is defined as a change in the distribution of X and y relative to the initial time t after a sufficiently long time Δt:

$$P_t(X, y) \neq P_{t+\Delta t}(X, y).$$

An example of the drift of the system of concepts is the introduction of US sanctions against independent countries. The introduction of such sanctions led to the strengthening of the popularity of the CNY (chinese yuan, growth in sales) as an international currency, and the countries of Europe - to a significant loss of negotiability (decrease in the number of contracts) and popularity, as well as a partial loss of demand for their products (decrease in income). To analyze changes in input datasets, a time-ordered sequence of their instances is required. A distinction is made between drift in streaming data and drift in packet (accumulated) data.

Drift most often switches to learning streams. The input data stream is defined as a continuous, unlimited sequence of data with an accepted time scale. This is different from dataset drift in a batch training program where the data is fully present in memory and processed all at once.

The following designations are used to study the types of drift of a system of concepts (Table 1):

TABLE 1. Mathematical notation

y, y+ , y-	target marks, respectively: positive together with negative, positive, negative
$P_t(X)$	is the probability distribution of the input data immediately after training the ML model
$P_t(y)$	preliminary probability distribution of target labels
$P_t(y\|X)$	is the posterior probability distribution of target labels
$P_t(X\|y)$	is the probability density distribution depending on the class

Here, the target labels y+ are true positive predictions (True Positive) together with false positive predictions (False Positive), and y- are true negative predictions (True Negative) together with false negative predictions (False Negative) from the error matrix (Fig. 2).

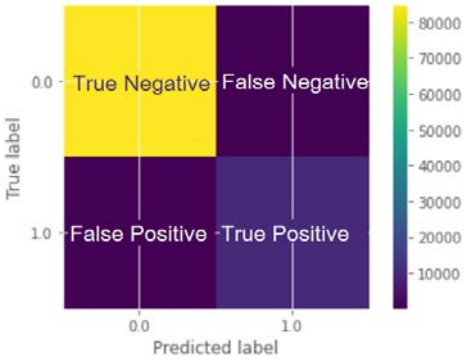

FIGURE 2. Confusion matrix

IV. Variety of Drift

If the set of drift functions D is defined on two arguments P_t and $P_{t+\Delta t}$, then the drift functions of the concept D_i (P^i_t , $P^i_{t+\Delta t}$) can be limited in their diversity by the properties presented in Table 2. For readability, the superscripts i of the probability distributions in the table are omitted.

On the set of given drift functions, the following definitions can be given to the types of conceptual drift $D_a - D_m$:

$D_a = D_1$, $D_b = D_2 \,\&\, D_4 \,\&\, D_5$, $D_c = D_3 \,\&\, D_4 \,\&\, D_5$,
$D_d = D_2 \,\&\, D_3 \,\&\, D_4 \,\&\, D_5$, $D_e = D_5 \,\&\, D_6$, $D_f = D_7 \,\&\, D_8$,
$D_g = D_{10} \,\&\, D_{11}$, $D_h = D_{12}$, $D_i = D_{13}$, $D_k = D_1 \,\&\, D_4$, $D_m = D_{14}$.

Next (Fig. 3) is a graphical representation of the various types of conceptual drift $D_a - D_m$.

V. Concept Drift Detection Methods

Let's consider the most effective existing methods for applying the new Data Mining lifecycle architecture.

A. Sliding Window

This method uses a fixed size window that "slides" over the dataset, starting with the oldest history. The task of such an analysis is to identify patterns of data drift. In this case, the use of triggers is typical, i.e. analytical software functions that track changes in the data specified by the data scientist or knowledge engineer. If such a trigger detects a programmed change, the new data becomes the source for training and testing the model, which then replaces the old model.

B. Trigger Ensembles of Model

Here we consider ensembles of machine learning models, each of which has its own triggers for determining when the model is updated by training on newly received data. In addition, it becomes possible to select from the ensemble those models that have retained the accuracy of classification or regression analysis. The ensemble metamodel can use an adaptive rule to combine models. The combination methods are predetermined by the knowledge engineer and can have a very complex structure.

C. Covariant Shift

A covariant shift is understood as a change in the ratio between the primary features involved in the training of the machine learning model. In this case, we can talk about a bias in the selection of features. Such a bias occurs either due to an initially systemic error in the selection of data for training, or due to the real dynamics of re-weighing the importance of features for learning, or the presence of any features that were not taken into account earlier or new ones that have just appeared. In any of these cases, automation is subject to updating and re-weighting of features with the formation of a new sample for training a new model.

Be sure that the symbols in your equation have been defined before the equation appears or immediately following. Italicize symbols (*T* might refer to temperature, but T is the unit tesla). Refer to “(1),” not “Eq. (1)” or “equation (1),” except at the beginning of a sentence: “Equation (1) is ... .”

TABLE 2. Concept drift functions

Drift functions	Concept Drift Formula	Naming concept drift
D_1	$P_t(y\|X) = P_{t+\Delta t}(y\|X)$	*No drift*
D_2	$P_t(y^-\|X) \neq P_{t+\Delta t}(y^-\|X)$	*Real negative drift*
D_3	$P_t(y^+\|X) \neq P_{t+\Delta t}(y^+\|X)$	*Real positive drift*
D_4	$P_t(X) / \neq P_{t+\Delta t}(X)$	*Covariant drift*
D_5	$P_t(y) \neq P_{t+\Delta t}(y)$	*Prior drift*
D_6	$P_t(X) = P_{t+\Delta t}(X)$	*No covariant drift*
D_7	$\exists x \in X,\ P_t(y\|x) \in C_1 \,\&\, P_{t+\Delta t}(y\|x) \in C_2$	*Fragment drift from class C_1 to class C_2*
D_8	$\exists x \in X,\ P_t(y\|x) \in C_2 \,\&\, P_{t+\Delta t}(y\|x) \in C_1$	*Fragment drift from class C_2 to class C_1*
D_9	$\exists x \in X,\ P_t(y\|x) \neq P_{t+\Delta t}(y\|x)$	*Fragmentary drift in any direction*
D_{10}	$\forall x \in X,\ P_t(y\|x) \in C_1 \,\&\, P_{t+\Delta t}(y\|x) \in C_2$	*Full drift from class C_1 to class C_2*
D_{11}	$\forall x \in X,\ P_t(y\|x) \in C_2 \,\&\, P_{t+\Delta t}(y\|x) \in C_1$	*Full drift from class C_2 to class C_1*
D_{12}	$\exists x \in X,\ P_t(y\|x) \in C_1 \,\&\, P_{t+\Delta t}(y\|x) \in C_2 \,\&\, \exists z \in X,\ P_t(y\|z) \in C_2 \,\&\, P_{t+\Delta t}(y\|z) \in C_1$	*Fragmentary drift in different classes*
D_{13}	$P_t(X_1) \neq P_{t+\Delta t}(X_1) \,\&\, P_t(X_2) = P_{t+\Delta t}(X_2)$	*Virtual drift in different classes*
D_{14}	$(C_t = C_1 \cup C_2 \,\&\, C_{t+\Delta t}) = (C_1 \cup C_2 \cup C_3)$	*Conceptual evolution*

D. Personalization Drift Correction

Personalization implies the presence of some "user" as a data source for the training sample. A change in the user's interests may be associated with some global events, for example, the outbreak of war, an earthquake or other natural disaster, a disease pandemic, etc. In this case, it is very important to determine the reason for the change in user interests and to identify a possible correction of the model or data selected for training. As a rule, new training is required after this. The personalization drift may be associated with a change in the qualitative contingent of users. This is a situation where some users disappear altogether, as factors determining the source of data, and new users appear with completely different interests. Here it is very difficult to automatically identify such personalization drift, if it is not provided in advance by means of monitoring the stages of the life cycle of Data Mining technology.

E. Online Learning

The model is continuously retrained or retrained on newly incoming data. The method is applicable only when the input data flow is systemic and can be reliably estimated by a knowledge engineer or data scientist.

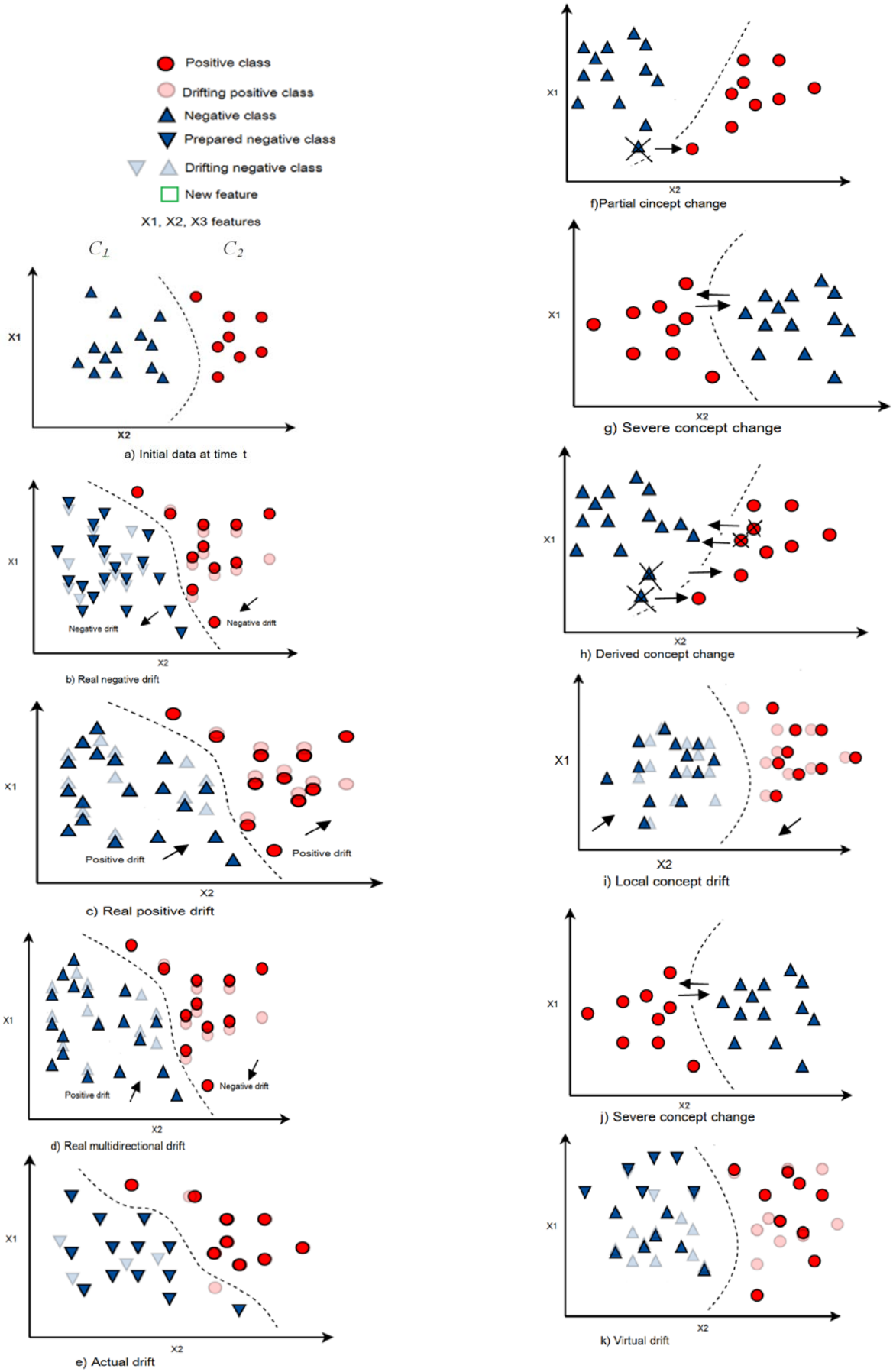

FIGURE 3. Types of drift concepts

The simplest correction method, in which the data drift is known in advance and is associated with certain time intervals. The intervals themselves can be identified depending on some additional conditions, but they are always exactly computable. This method can be used, as a rule, with frequent repetition of situations in the subject area of data analysis. The correct identification of the time period by its start and end points completely solves the issue of choosing a model from a pre-formed list.

F. Learning with a Teacher

The previous method approach developed using feedback. For this, reinforcement rules are formed that change the machine learning model in accordance with the incentives of the teacher (knowledge engineer). The teacher after each new classification or regression introduces a marker of the correctness of the prediction. The model must be designed in such a way that the reinforcement rules allow it to be corrected reliably.

G. Low Precision Sampling

The method consists in searching for those data sets on which the drift is most pronounced. A special sample is selected containing only cases of pronounced drift. Such samples are formed each time a drift is detected that occurs within (or after) a certain time. Low-precision samples are added to the original data set selected at the very beginning of the formation of the machine learning model. After a set of composite samples is formed, a new training, testing and validation of the model is performed.

H. Monitoring and Clipping Features

In the case of using an ensemble of models, the features of the input data set are monitored. If the same feature is used in different models of the ensemble and the value of the target variable remains unchanged, this feature for each of the models is studied on the ROC-AUC accuracy characteristic [19]. If a failure of the ROC-AUC value for this feature is detected below the threshold value specified by the knowledge engineer (for example, 0.81), the feature is considered to be drifting and is excluded from the dataset intended for training the machine learning model. The model is retrained on a new truncated dataset and then put into production.

VI. Semantic Fragmentation Method

The method of semantic fragmentation (MSF) proposed here by the author is to use knowledge models [20], in particular ontological models, to split input data sets into local fragments. The features of the input data set are digitized properties of concepts or relations of the knowledge model of the domain in which the task of classifying or predicting target values is. Thus, concepts or relations distinguish from the whole set of features local subsets that characterize these concepts or relations. If we use the genus-species and causal taxonomy [21, 22], local subsets of features will be nested subsets of each other. The more general a concept is in its semantics, the greater the number of elements included in the set of features corresponding to this concept.

If we divide the input data set into clusters, then taking into account the taxonomy of the concepts of the subject area, this can be done by classifying the data according to the specification of concepts from the bottom up or from the top down. It may happen that part of the features of one concept and part of the features of another concept are in the drift, and these two concepts do not have a common ancestor. This may indicate the situations listed below.

1. The taxonomy structure is incorrect.
2. There are not enough features for an accurate generic classification.
3. The taxonomy is not classified in sufficient detail.
4. There is an undefined concept of the generic hierarchy of contiguous inheritance. In the literature, contiguous inheritance is sometimes inaccurately defined as multiple inheritance or diamond-shaped inheritance.
5. The time frame is not defined precisely enough for two signs. I.e., in reality, the time of onset, continuation and end of drift is different for each of them.
6. Two signs of completely different concepts accidentally found themselves drifting together *in the same time frame*.

All of these situations in most cases require additional work with the domain data model. The subject area is described initially in the form of some effective knowledge model, for example, ontological, using OWL language tools [8]. Then this knowledge model undergoes fragmentation, involving the allocation of relatively independent fragments of knowledge corresponding to the semantic interpretation of the data of the input training set. Independent (local) fragments of the knowledge model are associated with the data of the input training set by the "concept - attribute" relationship. Individual concepts, having an appropriate list of attributes, represent a relatively independent semantic fragment.

If the domain knowledge model is designed correctly, the entire set of fragments completely covers the entire list of features of the input training dataset of the machine learning model. Now it is possible to check the signs for the presence of data drift without using a full search, since groups of signs united by a single paradigm are being investigated at once. At the same time, this does not exclude a consistent clarification of the role of each of the semantic fragment features in the identified data drift.

VII. Example of MSF Application

As an example, the subject area of communication services was investigated (Fig. 4). In it, the main concept is the concept of "Participants", which through genus-species relations can be a "Customer" of services or a performer "Provider", who, in the end, turn out to be specific people with known characteristics.

Figure 4 shows not only the main concepts of the service sector, but also individual features that can be represented in numerical terms: "Phone", "TV", "duration of use", "Orders", "Relevance", "gender". Now all numerical features can be fragmented according to the method described earlier according to the concepts defined by them (Table 4). For example, the concept of "Services" is simultaneously a semantic index for a group of features "Phone", "TV", "Internet". The concept of "Participants" is also interpreted as a semantic index of the group "Gender", "duration of use".

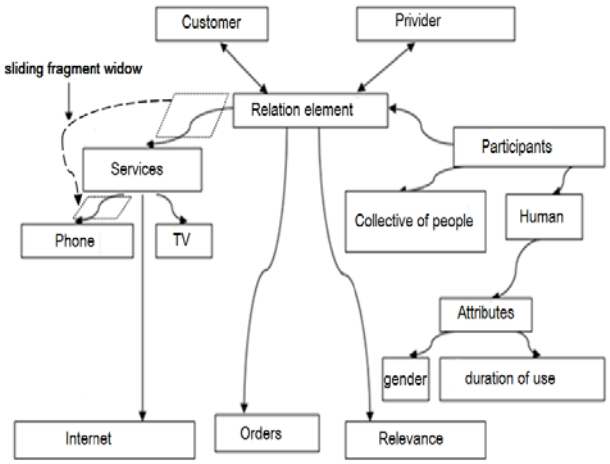

FIGURE 4. Fragment of the Architecture of the Ontology «Communication services» based on Semantic Fragmentation Method

TABLE 4. Fragment of the input data set

ID	gender	Rele-vance	Partner	TV	Du-ration	Phone	Multiple Lines	Internet Service
557	Male	0	No	No	34	Yes	No	DSL
366	Male	0	No	No	2	Yes	No	DSL
930	Female	1	No	No	8	Yes	Yes	Fiber optic

VIII. CONCLUSION

As shown by practical experiments [23], the greatest drift of concepts (ROC-AUC: 0.82 → 0.74 over two years) was detected for a group of features with the index "Participants". When updating these data in the training dataset, the accuracy characteristic of ROC-AUC with the help of MSF was practically restored (ROC-AUC: 0.74 → 0.80).

ACKNOWLEDGMENT

I would like to thank Vladimir Konov, data-scientist at «Regium Corporation, Ryazan, Russia», for practical support.

REFERENCES

[1] M. Asghari, D. Sierra-Sosa, M. Telahun, A. Kumar, A.S. Elmaghraby, Aggregate density-based concept drift identification for dynamic sensor data models, Neural Comput. Appl. 33 (8) (2021) 3267–3279.

[2] M. Tsiakmaki, G. Kostopoulos, S. Kotsiantis, and O. Ragos, Transfer learning from deep neural networks for predicting student performance, Applied Sciences, vol. 10, no. 6 (2020) pp. 2145–2218.

[3] Ch. Schröerab, F.Kruseb, J.M.Gómezb, A Systematic Literature Review on Applying CRISP-DM Process Model, Procedia Computer Science 181 (2021) 526–534.

[4] F. Bayram, B.S.Ahmed, A.Kassler. From concept drift to model degradation: An overview on performance-aware drift detectors. Knowledge-Based Systems 245 (2022) 108632

[5] J. P. Barddal, H. M. Gomes, F. Enembreck. Sfnclassifier: A scale-free social network method to handle concept drift, in: Proceedings of the 29th Annual ACM Symposium on Applied Computing, SAC '14, ACM, New York, NY, 41 USA, 2014, pp. 786–791. doi:10.1145/2554850.2554855.

[6] J. P. Barddal, H. M. Gomes, F. Enembreck. Sncstream: A social networkbased data stream clustering algorithm, in: Proceedings of the 30th Annual ACM Symposium on Applied Computing, SAC '15, ACM, New York, NY, USA, 2015, pp. 935–940. doi:10.1145/2695664.2695674.

[7] R. F. de Mello, Y. Vaz, C. H. G. Ferreira, and A. Bifet. On learning guarantees to unsupervised concept drift detection on data streams. Expert Syst. Appl., 117:90–102, 2019.

[8] C. Göpfert, L. Pfannschmidt, J. P. Göpfert, and B. Hammer. Interpretation of linear classifiers by means of feature relevance bounds. Neurocomputing, 298:69 – 79, 2018.

[9] I. Goldenberg and G. I. Webb. Survey of distance measures for quantifying concept drift and shift in numeric data. Knowl. Inf. Syst., 60(2):591–615, 2019.

[10] J.Gama, I.Žliobaitė, A.Bifet, M.Pechenizkiy, A.Bouchachia A survey on concept drift adaptation. ACM Comput. Surv. 46(4), 1–37, 2014.

[11] M. Khannouz and T. Glatard, "Mondrian Forest for Data Stream Classification Under Memory Constraints," 2022.

[12] M. Khannouz, B. Li, and T. Glatard, "OrpailleCC: a Library for Data Stream Analysis on Embedded Systems," The Journal of Open Source Software, vol. 4, p. 1485, 2019.

[13] J. L. Lobo, J. Del Ser, and E. Osaba, "Lightweight Alternatives for Hyper-parameter Tuning in Drifting Data Streams," in 2021 International Conference on Data Mining Workshops (ICDMW), 2021, pp. 304–311.

[14] A.Dehghani, O. Sarbishei, T. Glatard, and E. Shihab, "A Quantitative Comparison of Overlapping and Non-Overlapping Sliding Windows for Human Activity Recognition Using Inertial Sensors," Sensors, vol. 19, no. 22, 2019.

[15] M. Khannouz, B. Li, and T. Glatard, "OrpailleCC: a Library for Data Stream Analysis on Embedded Systems," The Journal of Open Source Software, vol. 4, p. 1485, 07 2019.

[16] H.M.Gomes, J.Read & A.Bifet. Streaming random patches for evolving data stream classification. In 2019 IEEE International Conference 1232 on Data Mining (ICDM) , 2019, pp. 240–249.

[17] Du, H., Zhang, Y., Gang, K., Zhang, L., & Chen, Y.-C. (2021). Online1197ensemble learning algorithm for imbalanced data stream. Applied Soft 1198 Computing,107 , 107378.

[18] Della Valle, E., Ziffer, G., Bernardo, A., Cerqueira, V., & Bifet, A. (2022).1183Towards Time-Evolving Analytics: Online Learning for Time-Dependent1184Evolving Data Streams. Data Science, p. 16.

[19] T. M. Ali, A. Nawaz, A. Rehman et al., A sequential machine learning-cum-attentions mechanism for effective segmentation of brain tumor, Frontiers in Oncology, vol. 12, pp. 1–10, 2022.

[20] F. S. Tsai, S. Cabrilo, H. H. Chou, F. Hu, and A. D. Tang, "Open innovation and SME performance: the roles of reverse knowledge sharing and stakeholder relationships," Journal of Business Research, vol. 148, pp. 433–443, 2022.

[21] K.Xia, Kai-Zhan Lee, Y. Bengio, E.Bareinboim. The causal-neural connection: Expressiveness, learnability, and inference. Advances in Neural Information Processing Systems, 34:10823–10836, 2021.

[22] Wang, T. D.; Parsia, B.; Hendler, J. (2006). "A Survey of the Web Ontology Landscape". The Semantic Web - ISWC 2006. Lecture Notes in Computer Science. Vol. 4273. p. 682.

[23] I.Yu. Kashirin, I.Yu.Filatov Formalized Description Of Intuitive Perception Of Spatial Situations. *2019 8th Mediterranean Conference on Embedded Computing (MECO)*, Budva, Montenegro, 2019, pp. 1-4.

Kashirin Igor Yurievich,
professor of the Ryazan State Radio Engineering University named after V.F. Utkin,
e-mail: igor-kashirin@mail.ru