Context–driven Data and Information Fusion

Vladimir Gorodetsky  
Practical Reasoning, Inc.  
New York, USA  
under the auspices of SPIIRAS  
St. Petersburg, Russia  
gor@practicalreasoning.com

Vladimir Samoylov  
Practical Reasoning, Inc.  
New York, USA  
under the auspices of SPIIRAS  
St. Petersburg, Russia  
samovl@practicalreasoning.com

Sergey Serebryakov  
Hewlett-Packard Laboratories  
St. Petersburg, Russia  
sergey.serebryakov@hp.com

Abstract—One of the major trends of modern data and information fusion technology is to take into account not only the fact of multiple sources but also the available context of a domain. An example case of the above statement is when data to be fused are represented in an object database and can be enriched by an expert with the domain ontology which provides a domain context to each instance of the data. The paper proposes context–driven data and information fusion technology. The latter comprises ontology-driven generation and aggregation of features as well as a two-step filtering of features: based on discriminative power at the first step and on a measure of causality at the second one. A peculiarity of the resulting solution is that every class of decision is specified by a specific set of features represented in terms of predicates that are statements about feature properties. The technology is implemented and validated in several applications of real life scale. In the paper, the technology is demonstrated by the example of applying it to the implementation of personalized intelligent MS Outlook e-mail assistant.

Key words–context, ontology, feature extraction, feature aggregation, feature filtering, causality.

1. INTRODUCTION

During the last decade, the application area of the data and information fusion technology is permanently spreading covering even larger number of modern application. The examples are the applications dealing with the (1) extraction of user–related information from heterogeneous information sources containing user's footprints; (2) e-trade customer profile mining for subsequent personalized recommendations; (3) social network mining intended for predictive detection of socially–dangerous communities within the homeland security context; (4) personalization of smart space services to the user's habits and preference, etc. However such applications put forward new requirements to the capabilities of the information fusion technology due to novel properties of the information to be fused. Fig. 1 outlines the types of available data sources and associated data and information properties peculiar to modern applications using fusion technologies.

It can be seen that footprints of a user can be found in many diverse sources, e.g., in internet search log, in logs of his/her access to web services and behavior in different internet shops. Rich information about user profile can be found in the content of his/her e-mail boxes and contact list, in the history of user's activity in social networks, etc. However joint processing of such data is a difficult task. Indeed, these data are unstructured and highly heterogeneous while containing numerical, categorical, ordered and textual data. They are of huge size and dimensionality and distributed over multiple sources. A great concern is data and information privacy.

An important issue of the fusion technology is to account multiplicity of contexts of the associated data, e.g., user's professional activity, family and cultural issues, political conviction, hobbies, etc. In fact, particular data and information context plays the topmost role in semantic understanding of the data and information fusion results. It can be extracted if to correctly and fully account available domain knowledge.

One of the rich domain knowledge sources making it possible to extract context of particular data and information instance is domain ontology. Indeed, domain ontology is meta-knowledge enriching each data record with additional concept, structure and attribute instances associated with these data inferred from ontology. It is worth to note that there exists an efficient middleware that is capable to in-fly transform a relational data sample with ontology on top of the
latter to the object form. An example of such middleware is Hibernate [7]. Therefore getting an object form of a relational data with ontology on top of them is an efficiently solved task.

However, context-driven data fusion puts forward new challenges. One of difficult of them is caused by the fact that in the object form, different instances of the same object can be specified in terms of different sets of concepts and attributes, i.e. in terms of different feature sets. Moreover, the same attributes (and features) in different object instances can be structured differently, thus representing various contexts within which particular object instances can exist. Let us note that this is the reason why the proposed fusion technology is called context-driven, or context-dependent.

The paper objective is to present the developed context-driven data and information fusion technology. Sections 2, 3 describe the basic ideas and foundation of this technology comprising feature generation, aggregation and two-step feature filtering based on measures of feature discriminative power, at the first step, and causality, at the second one. Section 4 describes a validation case study that is personalized intelligent MS Outlook e-mail assistant. Conclusion sketches the paper results and future efforts.

2. CONTEXT-DRIVEN INFORMATION FUSION

A. Problem statement

The statement of the problem considered in the paper assumes that distributed data and information are represented in the object form, i.e. in the relational database with the domain ontology on top of it. Otherwise the preliminary step of the fusion technology should be design of the domain ontology. Let us remind that object database is usually a virtual notion because each instance of the aforesaid object data is formed "on the fly" using an existing software (e.g. Hibernate [7]) in response to an inquiry to the distributed relational database if the inquiry is represented in terms of the ontology concepts. There is no limitation imposed on the data and information types: they can include numerical, categorical, ordered, textual, etc. data types.

It is assumed that every instance of the (distributed) object is assigned a label \( \omega_k \in \Omega = \{ \omega_1, \omega_2, \ldots, \omega_q \} \) indicating its class. In general case, class labels \( \omega_k \) are categorical or ordered values. For example, class labels may indicate a measure of a customer's interest to buy a particular type of product. In this case it is of ordered data type.

B. What is context-dependency? Case of intelligent MS Outlook e-mail assistant

Context is the data and information about particular environment within which a concept or an object exists. It includes information about a concept that is implicitly available through the ontology-based relations between concept in question and other ontology concepts. With regards to data and information fusion process, the usage of context makes it possible to include in the fusion algorithm additional semantically meaningful information thus improving the quality of the fusion result.

Let us explain the case study of an intelligent MS Outlook e-mail assistant and how context information is extracted and used for learning the fusion strategy. The objective of an e-mail assistant is to recommend to the user into which folder each incoming e-mail should be placed. E-mails contained in the user's structured folders, contact list and other information (described below) are utilized as an object data sample to be used for training and testing the assistant to fuse newly incoming data and information. In this case, the overall quality of recommendations to be produced by the trained

![Figure 2. Example of e-mail instance.](image-url)
assistant will depend on the context involved in the fusion procedure.

Fig. 2 shows an e-mail example. One can see the diversity of information types contained in the e-mail object: e-mail formal properties (importance, sender and receiver, etc.), subject, text of the e-mail body with the associated history of re-writing, attachments. Valuable information can also be additionally obtained indirectly through the use of ontology. For example, the ontology of an e-mail assistant (Fig. 3) provides additional information about persons, companies, e-mail addresses etc. mentioned in the e-mail body and/or its subject. This indirect information is exactly what is usually called the context of an e-mail instance since it is not explicitly contained in the e-mail itself.

Let us outline the e-mail assistant ontology, which UML diagram is depicted in Fig. 3. It comprises of two parts. The first part specifies the structure of the high level ontology concept EMailItem. The second one specifies the concept Person. The latter is used to represent information about e-mail box owner as well as about any other person mentioned in the e-mail contact list.

Information about Person may be very different. In our case, Person may occupy a single Position in a Company or several Positions in different companies. Company has its own postal address PostAddress, phone number Phone, web domain WEBDomain and web address WebURL which are indirect properties of the e-mail account owner Person or other ones mentioned in contact list. Person is also assigned InstantMessengerUIN list representing addresses (logins) of personal messengers and nicknames in social networks, e.g. addresses in Skype, ICQ, QIP, Yahoo-messenger, nicknames in Facebook, Live Journal, etc. All the data associated with the contact list Contact should be imported into object database. This information can, e.g., contain companies’ names and descriptions of the companies and their products, some specific information about particular persons, etc.

The concept EMailItem is also enriched by contextual information. Except formal properties, its components are e-mail body BodyContent and subject EMailSubject that are the most informative parts of any e-mail instance. Specific attribute of an e-mail instance is the name of the folder in which it is placed. For every new e-mail this attribute is unknown: it is the subject of the recommendation to be produced by the intelligent e-mail assistant. E-mail instance may be attached with one or more files Attachment having attribute AttachmentType.

C Secondary features and text processing: Case of intelligent MS Outlook e-mail assistant

Some other properties of the e-mail that seem to be important from domain viewpoint can be additionally introduced as concepts of the ontology, according to an expert initiative. They are called secondary features. In the MS Outlook e-mail assistant case study, they are used to represent important implicit information contained in the e-mail body and/or in the e-mail subject. For the case study in question, UML diagram of the secondary features is shown in Fig. 4. For example, the name of a company may become a secondary concept if its phone number is found in the e-mail body. In such case, the company name is considered as a concept connected to the e-mail body. Some of them are auxiliary (e.g. RelatedItems of BodyContent, EMailSubject, Attachment, etc.), whereas others are to be involved in learning process (e.g. key words, e-mail addresses, phone numbers, instances of InstantMessengerUIN, people’s/company names, etc.).

The secondary features are usually extracted using some specific software intended for semantic processing of a text represented in terms of a natural language. In particular, text analysis software used in MS Outlook prototype plug-in is based on two IBM tools: IBM Language Resource Ware (LRW) [5] and IBM Ontological Network Miner (ONM) [6] both available at IBM’s alphaworks site [14]. Mechanism of regular expressions is used to extract web and e-mail

Figure 3. Domain ontology of e-mail assistant.
addresses. LRW is used for processing the texts in natural language. It can be used to configure the processing pipeline that includes such components as language identifier, sentence splitter, tokenizer, named entity recognizer, etc. In particular, the following LRW capabilities are used:

1. **Dictionary–based annotator** is used to extract from a text variety of named entity types like persons' and companies' names, etc. It is based on dictionaries containing necessary words.

2. **Rule–based annotator** is used to extract higher level entities based on extraction rules. For instance, in the text fragment *Barry White* the word *Barry* would be annotated as *FirstName* using dictionary of people names and *White* would be annotated as a capitalized word (*CW*) during the tokenization step. Then the rule "*FirstName* CW" (this rule matches text fragments containing first names followed by capitalized words) would tag the entire text (*Barry White*) as *FullPersonName*.

Once the text processing pipeline is configured, LRW can create and deploy UIMA\(^1\)–compliant analysis engines to use the text analysis components in the applications dealing with unstructured data. The motivation of using dictionaries and rules for information extraction in this particular application is that (1) they are extremely fast comparing to other approaches based on statistical machine learning and (2) rules are quite suitable for extracting such complex structures as phone numbers, postal addresses, e-mail and web addresses, etc.

ONM [6] is used for more sophisticated analysis, e.g. to extract key concepts from the text (*text focus*), even those presented in the text implicitly. ONM can disambiguate the concepts. It uses ontology describing concepts and relations between them. The ontology used by ONM in e-mail assistant software prototype was formed using categories and concepts as well as relations between them derived from Wikipedia dump.

Summarizing this section, the text analysis is done in two steps: (1) extraction of concepts (found in the text) using dictionaries and rules and (2) detecting key concepts (related items) using ONM.

Fig. 5 illustrates ontology-based (object) representation of the instance of e-mail given in Fig. 2.

### 3. THEORETICAL BASIS

#### A. Fusion technology "philosophy"

The developed context-driven information fusion technology is based on the following quite natural "philosophy":

1. **Ontology and features.** Every feature should be expressed in terms of ontology notions and/or their attributes thus providing it with a well understandable semantics. If an expert wants to introduce a feature that is not explicitly contained in the domain ontology he/she has to take care about establishing its relations to the existing ontology concepts.

2. **Features and classifiers.** There is no semantic difference between the concept *feature* and the concept *classifier*. Every feature can be considered as a one–feature classifier as follows, for example:

   \[
   \text{if } F(X_i \in X) \text{ then } \omega_k \tag{1}
   \]

   where \( F(X_i \in X) \) is an unary predicate and \( X_i \) is a subset of domain of the feature \( X \), and vise versa, every classifier can be interpreted as a feature, e.g. \( F(X_i \in X) \).

   This means that use of multiple classifier fusions has no semantic difference in comparison with conventional classification on the basis of multiple features.

3. **Good and bad features:** Slightly re-formulating the Condorcet theorem [16] one can say that a classifier is "good" if its accuracy is strictly more than 0.5. Otherwise a classifier is bad. Analogously, one can say that a feature is "good" if a one-variable classifier (1) using this feature is "good".

4. **Main receipts:** The "recipes" against huge scale of both data size and dimensionality are feature aggregation and feature filtering using a threshold value of a feature usefulness measure.

5. **Personalization:** Feature selection procedure should be class–targeted, i.e., a specific set of features are generated for each class of object instances.

The developed fusion technology strictly follows these principles. Let us outline it in step-wise manner.

#### B. Feature aggregation

It is assumed that an initial feature set is generated by an expert and each feature of this set is expressed in terms of the ontology concepts and/or their attributes. The total cardinality of the initial feature set can be arbitrary. The feature set should be clearly understandable and semantically well interpretable.

The basic idea of aggregation step is to introduce for each class \( \omega_k \) and each feature \( X_i \) defined over some domain a

---

\(^1\) UIMA (Unstructured Information Management Architecture) framework is an open platform for building applications or search solutions that process text or other unstructured information to find the latent meaning, relationships and relevant facts buried within (http://uima-framework.sourceforge.net/).
limited number of useful predicates (statements about feature properties) in the form \( F(X_i, X_i^{(k)}) \) that are true over domains \( X_i^{(k)} \). Let us explain how these predicates are introduced.

Let us use some formal criteria to split the domain \( X_i \) of a feature \( X_i, \ i \in \{1, ..., n\} \) into disjoint subsets \( X_i^{(k)} \subseteq X_i, \ k = 1, ..., m \), such that each \( X_i^{(k)} \) is assigned a class label \( \omega_k \in \Omega, \ \Omega = \{\omega_1, \omega_2, ..., \omega_q\} \). There is no requirement \( \bigcup_{k=1}^{m} X_i^{(k)} = X_i \). Each set \( X_i^{(k)} \) is such that it contains those domain values of the feature \( X_i \) that are more frequently met in the object instances of the class \( \omega_k \) than in any other class. In other words, for each particular feature \( X_i \), the rule "if \( X_i^{(k)} \subseteq X_i \) then \( \omega_k \)"., \( k = 1, ..., m \), is considered as a simple one-feature classifier that is capable to produce a correct decision with the probability \( p(\omega_k / X_i^{(k)}) \), and this probability is larger than the probability \( p(\omega_i / X_i^{(k)}) \) for any other class \( \omega_i \neq \omega_k \). In fact, this approach implements one–feature Naïve Bayesian classifier. Indeed, the aggregation procedure assumes that any particular value \( a_i \in X_i, \ i \in \{1, ..., n\} \) belongs to the domain \( X_i^{(k)} \) \( (a_i \in X_i^{(k)}) \) if and only if\
\[
p(\omega_k / a_i) > p(\omega_i / a_i) + \Delta, \text{ for all } \omega_i \neq \omega_k. \tag{2}
\]

In formula (2) \( p(\omega_k / a_i) \) is the conditional probability of the class \( \omega_k \) under condition \( X_i = a_i \), \( p(\omega_i / a_i) \) is the conditional probability of any other than \( \omega_k \) class under the same condition, and \( \Delta \) is a positive number having the sense of a threshold. It is worth to note that the procedure computing the aggregates \( X_i^{(k)} \) presented in the formula (2) is efficient since the estimates of the aforesaid conditional probabilities are quickly computed using object database.

Next step of the aggregation is the introduction of the set of the unary predicates \( F_i^{(k)}(a_i) \) corresponding to the aggregates \( X_i^{(k)} \) in the following way. Each predicate \( F_i^{(k)}(a_i) \) is assigned the truth value \textit{true} if and only if \( a_i \in X_i^{(k)} \). Therefore \( X_i^{(k)} \) is the domain where the unary predicate \( F_i^{(k)} \) takes the value \textit{true}. Next, for each unary predicate \( F_i^{(k)} \), the conditional probability \( p(\omega_k / F_i^{(k)}) \) of the class \( \omega_k \) given that the predicate \( F_i^{(k)} \) takes the value
true is simply computed using the database of object instances. Thus, the result of the aggregation procedure consists of the set of the predicates \( F_{i}^{(k)} \) \((i \in \{1, \ldots, n\}, k \in \{1, \ldots, m\}\), forming the initial list of aggregated features assigned
   i) the domains \( X_{i}^{(k)} \) where they take the values true and
   ii) conditional probabilities \( p(\omega_{k} / F_{i}^{(k)}) \).

The attractive properties of the aggregated features \( F_{i}^{(k)} \)

\[
p(\omega_{1} / F_{1}^{(1)}) \quad , \quad p(\omega_{2} / F_{2}^{(2)}) \quad , \quad \ldots \quad , \quad p(\omega_{i} / F_{i}^{(i)}) \quad , \quad \ldots \quad , \quad p(\omega_{m} / F_{m}^{(m)})
\]

(4)

\[
(\omega_{k} \in \{+, \star, \bullet\} \quad , \quad k \in \{1, \ldots, m\} \quad \text{are at least twofold:}
\]
i) independently of the initial measurement scales of the features \( X_{i} \), the aggregated features \( F_{i}^{(k)} \) are measured in Boolean scale. Therefore, object database, in this feature space, is transformed to homogeneous (Boolean) scale thus significantly simplifying the data and information fusion problem, and
    ii) context—dependent data of the object database are transformed to the set of predicates assigned truth values.

Fig.6 illustrate aggregation step by an artificial example. In fact, this figure is self-explanatory. It graphically demonstrates how \( X_{i}^{(k)} \) are aggregated, where \( k=\{+, \star, \bullet\} \) are the class labels and the feature set is \( \{X_{1}, X_{2}\} \).

C. Discrimination power-based feature filtering—I

The result of the aggregation procedure is the set of predicates \( F_{i}^{(k)} \) with the given truth domains \( X_{i}^{(k)} \), marked by the class index and assigned conditional probabilities as indicated below:

Feature filtering procedure consists in two steps. At the first filtering step, each such feature is used to build one-feature Naïve Bayesian classifier. The latter have to be tested in order to compute contingency matrices diagonal probabilities \( p(\omega_{k} / F_{i}^{(k)}) \) and \( p(\omega_{k} / -F_{i}^{(k)}) \) with the subsequent filtering them according to the following rule:

If \( p(\omega_{k} / F_{i}^{(k)}) + p(\omega_{k} / -F_{i}^{(k)}) > 0.5 \) then the feature \( F_{i}^{(k)} \) is preserved, otherwise — filtered.

The result of this step is the set of "good" features that can also be considered as "good" classifiers, according to the sense introduced in section 3.A.

D. Casual analysis-based feature filtering—II

Since the time when Bayesian network [10] and subsequent causal Bayesian networks [11] have made it possible to detect causal structures in data, the latter are considered as a convenient framework for many applied decision making problems where it is necessary to reason about causality. In particular, since 1990-th, causal Bayesian networks is considered as a powerful approach to causality—based discovery of informative features. The basic idea behind causal feature discovery is to find a subset of data attributes that are strongly correlated with the class label and to test, afterwards, which of them actually correspond to causal relation [2, 3, 12]. Later on, the idea of causal feature discovery using Causal Bayesian Networks was developed in depth [1, 13, etc.]. At present days this idea is very popular in data mining. Unfortunately, use of correlations as preliminary step and subsequent testing, e.g., using some statistical test [12] leads to its restricted efficiency and inapplicability to huge data dimensionalities causal analysis. In the developed causality analysis technology, a measure for direct estimation of the causality between a feature and class label is proposed. It results in significant reduction of computations. Let us sketch it.

Let \( F_{i}^{(k)} \) be an aggregated feature selected in filtering–I procedure as a candidate feature for class \( \omega_{k} \) for further investigation. Let us use as a causality measure the following function:

\[
R(F_{i}^{(k)}, \omega_{k}) = p(\omega_{k} / F_{i}^{(k)}) - p(\omega_{k} / F_{i})
\]

(4)

This function is known, in probability theory, as regression coefficient of two random events (not random variables), \( R(F_{i}^{(k)}, \omega_{k}) \in [-1, 1] \). This relation is not symmetric and if \( |R(F_{i}^{(k)}, \omega_{k})| > |R(\omega_{k}, F_{i}^{(k)})| + \Delta \) then feature \( F_{i}^{(k)} \) is considered as a cause for the class with label \( \omega_{k} \), where \( \Delta \) is a positive threshold value determining the "strength" of causality.

For practical needs, the value of regression coefficient \( R(F_{i}^{(k)}, \omega_{k}) \) can be computed as follows:
All probabilities estimates can be simply computed on the basis of object database transformed into the space of aggregated features \( F_{i}^{(k)} \). Thus, the causality-based feature filtering rule to be used is as follows:

\[
R(F_{i}^{(k)}, \omega_{k}) = \frac{[p(\omega_{k}, F_{i}^{(k)}) - p(\omega_{k}) \times p(F_{i}^{(k)})]}{[p(F_{i}^{(k)}) \times (1 - p(F_{i}^{(k)}))].} \tag{5}
\]

All probabilities estimates can be simply computed on the basis of object database transformed into the space of aggregated features \( F_{i}^{(k)} \). Thus, the causality-based feature filtering rule to be used is as follows:

\[
| R(F_{i}^{(k)}, \omega_{k}) | \geq \delta. \tag{6}
\]

where \( \delta \) is a positive threshold value.

4. EXPERIMENTAL RESULTS

The proposed approach to data and information fusion was implemented as a domain-independent reusable core and domain specific software components what makes it possible to employ it with a little additional effort to a wide range of diverse applications. The general components of the architecture of the developed software and their interactions are depicted in Fig. 7. At the figure, the rectangles correspond to the software components whereas other blocks present data and information supporting the system operations. The dashed arrows indicate the data flows using in learning processes circle whereas other ones indicate the data flows of the fusion procedure.

Validation was done using several applications. Let us outline some experimental results relating to the intelligent MS Outlook e-mail assistant, which data, information and knowledge sources were described in section 2.

Experiments were conducted for a real-life user e-mail mailbox including 20 folders (organized in a hierarchical manner) each containing decades of e-mails. The learning data (e-mails) were divided into two sets – training and testing. Experimental setup was as follows:

1. **Attributes of aggregation and filtering algorithm.**

For categorical data aggregates (both single and pair-wise) the threshold value of \( \Delta = 0.4 \) was used in the aggregate forming rule (2). Aggregates for numerical features were formed using modified Quinlan measure \([9]\) with numerical domain quantized into 10 unequal subintervals.

2. **Setting of causality discovery mechanism.**

The threshold \( \delta \) in causality-based filtering rule (6) was changed gradually in order to generate 30 best causes that can potentially be used as the premises of the rules

\[
\text{if } F_{i}^{(k)} \text{ then } \omega_{k} \text{ for all } F_{i}^{(k)} \text{ and } \omega_{k}. \tag{7}
\]

It is worth to note that, in these experiments, the rules discovered were more sophisticated than presented in (7). In fact, one more step of the features generalization was implemented in order to discover the rules which premises are formed as conjunctions of the discovered causes \( F_{i}^{(k)} \) for each class label. A special discovery mechanism was developed for this purpose. Due to limited paper space and due to the fact that the essence of such mechanism is out of the paper scope, information about this mechanism is omitted here.

3. **Decision making.**

As a decision making mechanism, a variant of combining of multiple classifiers decisions was implemented, in which the cause–consequence rules play the roles of classifier ensemble. Some results are given in Tab. 1. Let us remind that experimental data corresponds to real–life mailbox, where data are rather complex and no simplifications were done. Experimental results are promising. Indeed, analysis of the results (Tab. 1) shows that, in general, the proposed fusion technology works well for the most of the folders; although decisions produced in regard to folders 5 and 20 are rather

<table>
<thead>
<tr>
<th>1. Accuracy averaged over the set of folders</th>
<th>Coverage</th>
<th>False Alarm</th>
<th>Refusal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probabilities</td>
<td>0.75</td>
<td>0.0833</td>
<td>0.167</td>
</tr>
<tr>
<td>Number of e-mails</td>
<td>9</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Accuracy related to particular folders</th>
<th>Coverage</th>
<th>False Alarm</th>
<th>Refusal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Folders’ ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0.5</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>0.5</td>
<td>0.5</td>
<td>0</td>
</tr>
</tbody>
</table>
bad. This fact does not falsify the technology itself. On the one hand, the number of samples in these folders is rather small. On the other hand, analysis of the particular e-mails for which the produced decisions are bad can prompt the needed ontology corrections and, perhaps, the necessary extension of the secondary feature set. With increasing of the learning sample size the accuracy of the e-mail classification should also be improved.

5. CONCLUSION

The paper contribution is the context–driven data and information fusion technology and reusable software engine that can be used in a wide range of diverse types of applications when decision is built on the basis of multiple distributed sources of heterogeneous data and information. The novelties of the proposed technology are threefold:

(1) proposal to enrich learning data sample with expert knowledge (i.e. ontology) and to transform this sample to an object data base representation makes it possible to take into account the context and the semantics of the available data at both learning and data and information fusion phases;

(2) automatic procedure of aggregation, selection and filtering the informative features; and

(3) the approach to transforming the entire heterogeneous data sample to homogeneous Boolean data representation that reduces the learning phase of the data and information fusion procedure to a conventional binary learning task.

The described technology is fully implemented and validated in several domains. Among them, intelligent MS outlook e–mail assistant, movie recommending system based on NetFlix data set [8], vibroacoustic data [4] and some others.

Further research is to enrich the text analysis and text mining technology to achieve better reusability of the software components and to validate the developed engine on a wider range of applications. Significant efforts will also be paid to improve the reusability of the software core implementing the basic operations of the developed technology. Further testing of the developed context-driven data mining and information fusion technology will also be

ACKNOWLEDGMENT

Participation in the Conference and travel to Singapore was supported by Office of Naval Research Global.

REFERENCES


