Inference Factor Identification Technology for Explaining Inference Result Made by Deep Tensor

● Tatsuru Matsuo ● Yusuke Oki ● Koji Maruhashi

Inference results provided by AI require accountability in terms of the reasons and basis behind the inference. For AI to be accepted in areas where accountability is needed, such as the medical and financial sectors in particular, the reasons and basis for an inference must be shown to earn sufficient trust. Unfortunately, explaining the reasons or basis for an inference is difficult for many of the AI methods that provide highly accurate inferences, such as deep learning. Deep Tensor, an AI technology developed by Fujitsu Laboratories, is capable of highly accurate analysis of graph data representing connections between people and things. With Deep Tensor, accounting for the reasons behind inferences is still an important issue. Accordingly, Fujitsu Laboratories has developed inference factor identification technology as a means of resolving this issue. The technology uses feature values called core tensors generated by Deep Tensor to indicate which elements of graph data contributed to the results of an inference, thereby providing an explanation. This paper describes inference factor identification technology and presents examples of its application in the medical and financial sectors.

1. Introduction

Advances in AI over recent years have led to its application in a wider range of fields. Above all, deep learning has been particularly remarkable and it has become more widely adopted, especially in the handling of image data. Unfortunately, because many of these advanced AIs function as black boxes, explaining the reasons behind their decisions is problematic. Fujitsu Laboratories developed Deep Tensor[1)-3)], a technology that is capable of the highly accurate analysis of graph data representing connections between people and things. However, explaining the reasons behind decisions made by Deep Tensor still posed a problem.

In response, Fujitsu Laboratories developed a technology for identifying the factors behind the inference result made by Deep Tensor. This inference factor identification technology can specify which elements in graph data contributed to inference result made by Deep Tensor. By doing so, this helps data experts to decide whether they can rationally interpret the inference result made by Deep Tensor, and therefore whether or not they can trust it. If a rational explanation of inferences can be given and the results trusted, then the practical use of AIs that operate on graph data becomes feasible even in fields like medicine and finance that demand accountability for decisions.

This paper describes inference factor identification technology and presents examples of its use in medicine and finance.

2. Challenges in explaining inference results made by Deep Tensor

While technologies already existed for obtaining explanations of AI inference results, issues associated with Deep Tensor's use of graph data meant that these technologies could not be used.

This section describes an overview of Deep Tensor and the difficulties to be overcome when seeking to explain the inference results made by it.

2.1 Overview of Deep Tensor

Deep Tensor is a deep learning technology developed by Fujitsu Laboratories that works on graph data, where "graph data" means data representing the connections among objects such as people and things. The people or things being connected are called "nodes" and the links between them “edges.” The challenge when using learning on such graph data is to extract its important features, and Deep Tensor is a way of achieving this.

Figure 1 shows an overview of how Deep Tensor works. The graph data is represented as a mathematical structure called a tensor. It obtains highly accurate inferences (classification and regression) by using the proprietary technology of structure-restricted tensor decomposition to extract the important features from the graph data and then inputting these into a neural network. The graph data is represented in tensor form that indicates whether or not edges exist between each pair of nodes. This tensor data is decomposed into a core tensor and factor matrices. While this is what is generally known as tensor decomposition, Deep Tensor decomposes the tensor in such a way that the core tensor contains important features of the graph data. Specifically, it introduces a target core tensor that serves as a criterion for tensor decomposition, decomposing the tensor in such a way that the core tensor approximates the target core tensor.

This target core tensor is not something that is provided in advance, rather it is obtained by learning in conjunction with a neural network using a proprietary training technology. That is, because the target core tensor is trained to improve inference accuracy, the resulting core tensor ends up containing the important features of the graph data that contributed to the inference result. The features that turn out to be important vary depending on the data. For example, sometimes the partial structure of the graph data is important, and sometimes the overall structure is important.

Thanks to this technology, Deep Tensor is able to make decisions about graph data with high accuracy. Unfortunately, because it contains a neural network that remains a black box, this in itself is not enough to relieve the difficulty of explaining inference results. To overcome this, a technology that enables black-box AI inference results to be explained was adopted and applied to Deep Tensor.

2.2 LIME: A conventional technology for explaining inference results

Local interpretable model-agnostic explanations (LIME)[4] is a conventional technology for explaining the inference results made by black-box AIs. LIME obtains its explanations by attempting to approximate the original model using a linear regression model with input variables suitable for explanation.

Figure 2 shows how LIME works using the example of an AI that assesses whether or not an input image shows a chicken. The image data in question is first split up into small sections and multiple versions of the image are produced in which random regions have been masked out. The AI is then used to estimate for each of these masked images the probability that it shows a chicken. Next, a regression model is trained to output the AI inference result using each of the masked images, with binary variables indicating whether or not each region of the image was masked (0 = masked, 1 = not masked) being used as the regression model inputs.

This involves weighted learning based on the similarity between the binary variables for the original image to be explained and the variables for each masked image. The result is that the linear regression model comes to approximate the AI behavior with regard to images similar to the image in question. By

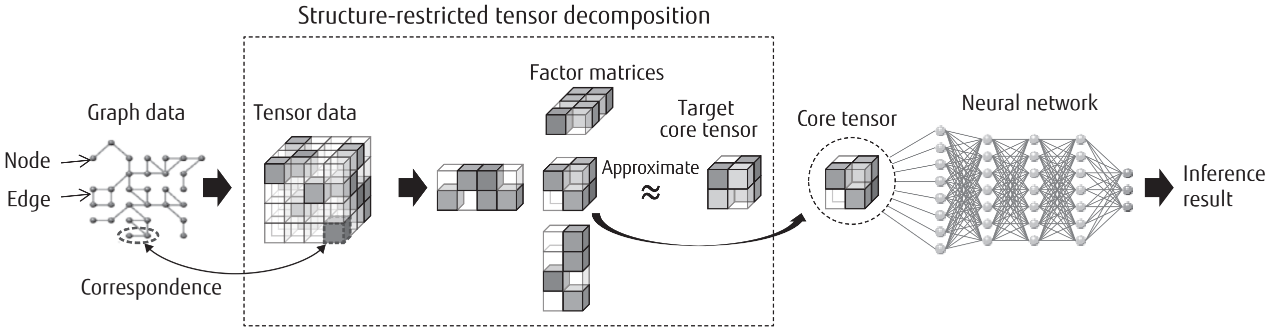


Figure 1
Overview of Deep Tensor.

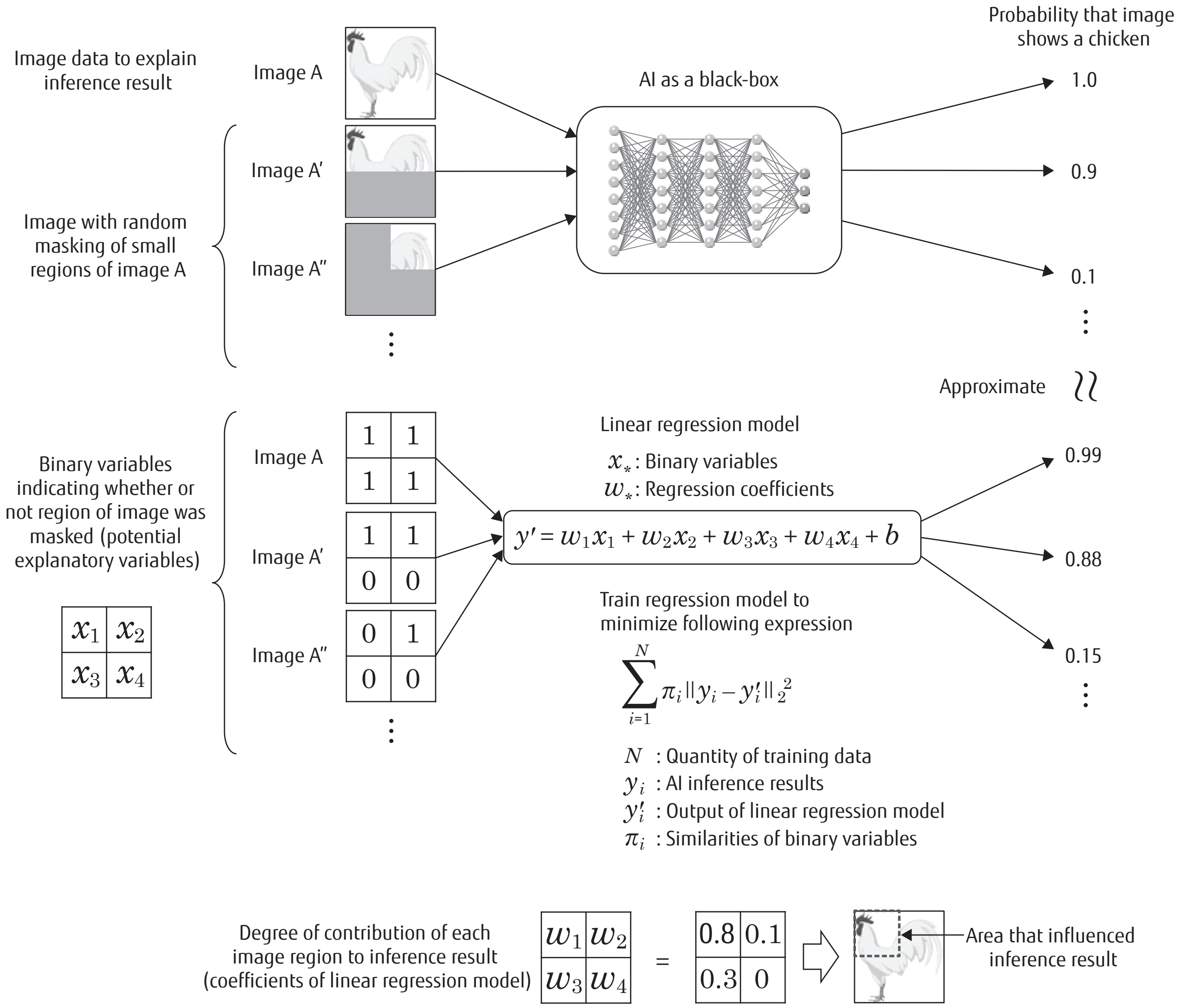


Image data to explain inference result
Image A
Image with random masking of small regions of image A
Image A′
Image A″
AI as a black-box
Probability that image shows a chicken
1.0
0.9
0.1
Approximate
Binary variables indicating whether or not region of image was masked (potential explanatory variables)
x_1 x_2
x_3 x_4
1 1
1 1
1 1
0 0
0 1
0 0
Linear regression model
x_* : Binary variables
w_* : Regression coefficients
$y' = w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + b$
0.99
0.88
0.15
Train regression model to minimize following expression
$\sum_{i=1}^{N} \pi_i \|y_i - y'_i\|_2^2$
N : Quantity of training data
y_i : AI inference results
y'_i : Output of linear regression model
π_i : Similarities of binary variables
Degree of contribution of each image region to inference result (coefficients of linear regression model)
w_1 w_2
w_3 w_4
=
0.8 0.1
0.3 0
Area that influenced inference result

Figure 2
Use of LIME in image analysis AI.

treating the regression coefficients for each variable in the resulting linear regression model as indicating its degree of contribution to the inference result, the assumption that those variables with a high regression coefficient made the greatest contribution to the inference result can be used to explain the result. In other words, they indicate which regions of the image in question played the largest role in inferring that it shows a chicken.

2.3 Challenges in explaining inference results extracted from graph data

When applying LIME to Deep Tensor, the challenge is to prepare variables that are suitable for explaining graph data. One method for approaching this challenge is to split up the graph data into small areas, as in the example of the image data described above. However, there is no obvious and appropriate way of splitting graph data. Moreover, even if an expert on the graph data concerned considers partitioning methods of graph data based on their expert knowledge, such a method cannot always be devised. Devising suitable partitioning methods from large data sets is particularly difficult. If an inappropriate partitioning method is used, the resulting inference explanations will also be of poor quality.

3. Inference factor identification technology

This section describes the technology used to identify the factors behind inference results made by Deep Tensor.

Similar to LIME, the technology trains a linear regression model to determine which elements in the graph data contributed to the inference results made by Deep Tensor. This involved using the mechanisms of Deep Tensor itself to overcome the issues noted in the preceding section.

As explained above, the core tensor generated by Deep Tensor contains the important features of the graph data. Therefore, if the input variable of the linear regression model in LIME is a core tensor, it eliminates the need to find a way to partition the graph data. In this case, training the linear regression model will show which elements of the core tensor contributed to the estimated results. Unfortunately, the core tensor is itself often difficult to interpret, making it difficult in such cases to explain an inference result in terms of the degrees of contribution obtained from the core tensor. Therefore, the degrees of contribution obtained for each element of the core tensor are transformed into the degrees of contribution for each element of the original tensor data i.e. the contribution of each edge in the graph data. Doing so indicates which parts of the graph data played an important part in the resulting inference.

Figure 3 shows a diagram of inference factor identification technology. First a linear regression model that outputs the Deep Tensor result using the elements of the core tensor as its inputs is trained in the same way as LIME. As the core tensor in the diagram example has eight elements, the regression model has eight inputs. This input variable corresponds to the x in the formula in Figure 3. The training data for this linear regression model consists of the core tensors of the graph data (hereafter, the target graph data) for which the inference results are to be explained, the core tensors of additional graph data, and the Deep Tensor inference results for these graph data. The "additional graph data" in this case could be the graph data used to train Deep Tensor, for example.

Training this linear regression model involves training the core tensors of the target graph data and graph data with similar core tensors to approximate the behavior of Deep Tensor. Specifically, the weight of every graph data point, including target graph data, is determined based on the similarity between the core tensors of the target graph data and the core tensors of all graph data.

Each regression coefficient in the trained regression model corresponds to an element in the core tensor and is assumed to represent the element's contribution to the inference result. Multiplying the factor matrices obtained by structure-restricted tensor decomposition by these contributions gives the contribution of each element in the original tensor data, which is to say the contribution of each edge in the graph data.

Comparing the contributions of each edge obtained above indicates which parts of the graph data had a large influence on the inference result. This facilitates rational interpretation of the result by a domain expert.

4. Example applications of inference factor identification technology

This section presents example uses of the technology in medical and finance applications.

4.1 Medical application

This example involves use of the technology in assessing the toxicity of chemical compounds in drug development. This example uses an open data set (Tox21[5]) to assess whether or not a compound is toxic based on its chemical structure.

The first step was to build the graph data for the compounds in the data set, treating each atom as a node and each inter-atomic bond as an edge, and to split this into separate data sets for learning and assessment, respectively. Next, Deep Tensor was trained using the learning data set and then run using the assessment data set. Inference factor identification technology was then applied to the results to assess the contribution of each inter-atomic bond to the toxicity inference. Finally, the bonds with a high degree of contribution were reviewed visually.

Figure 4 shows the chemical structures of two compounds identified by Deep Tensor as being toxic. The bold lines in the figure indicate those bonds that were determined by inference factor identification technology to make relatively large contributions to the result. Looking at which bonds in compounds A and B had a large contribution shows that the same chemical

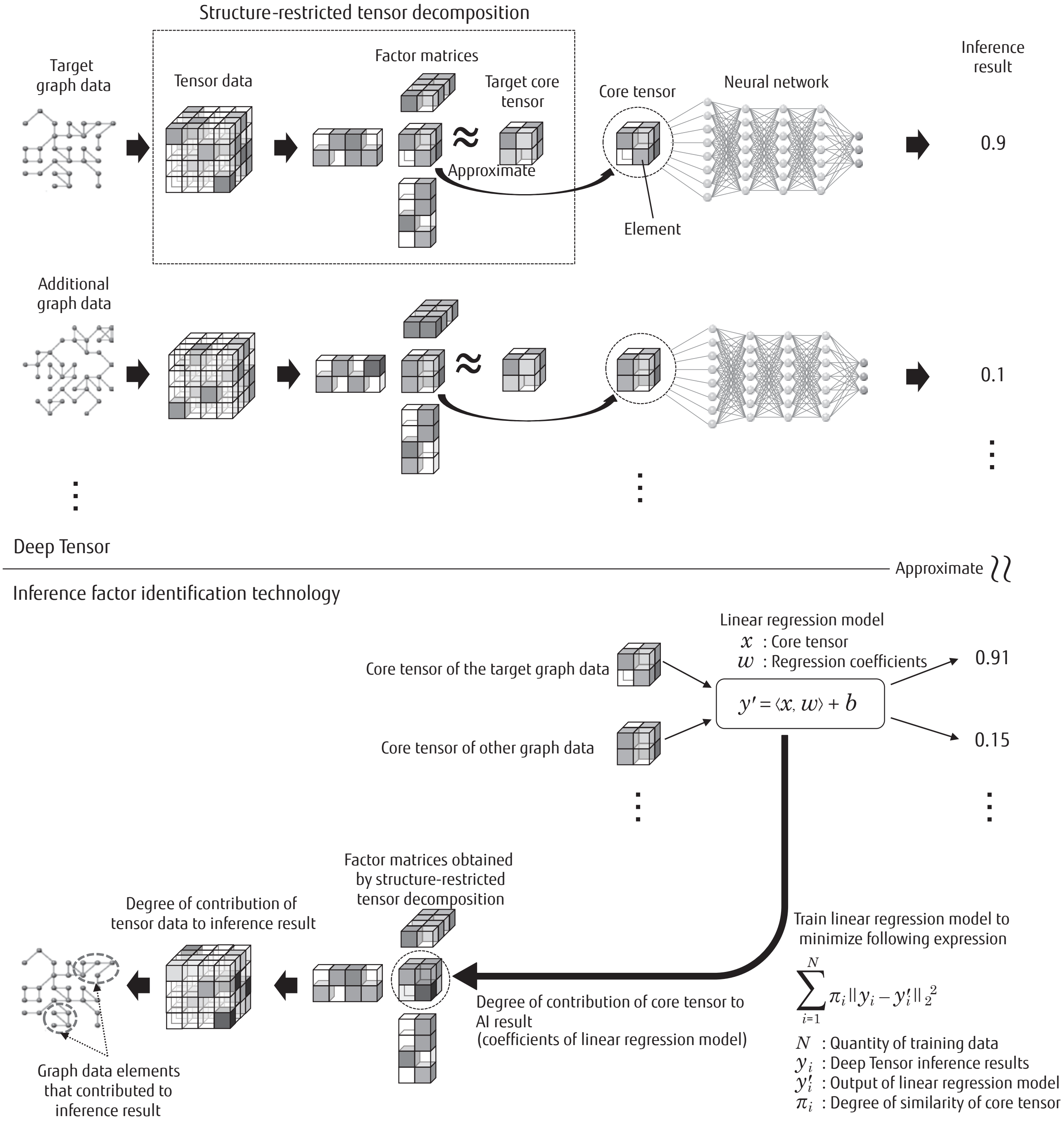
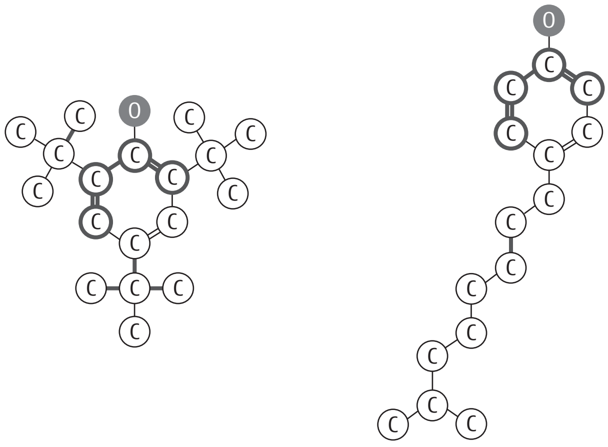

Structure-restricted tensor decomposition
Target graph data
Tensor data
Factor matrices
Target core tensor
≈
Approximate
Core tensor
Element
Neural network
Inference result
0.9
Additional graph data
≈
0.1
Deep Tensor
Approximate
Inference factor identification technology
Linear regression model
x : Core tensor
w : Regression coefficients
Core tensor of the target graph data
$y' = \langle x, w \rangle + b$
0.91
Core tensor of other graph data
0.15
Factor matrices obtained by structure-restricted tensor decomposition
Degree of contribution of tensor data to inference result
Train linear regression model to minimize following expression
$\sum_{i=1}^{N} \pi_i \| y_i - y'_i \|_2^2$
Degree of contribution of core tensor to AI result (coefficients of linear regression model)
Graph data elements that contributed to inference result
N : Quantity of training data
y_i : Deep Tensor inference results
y'_i : Output of linear regression model
π_i : Degree of similarity of core tensor

Figure 3
Overview of inference factor identification.

structure is involved in both cases.

These two compounds have similar core tensors in Deep Tensor. This indicates that similarities exist in the chemical structures that played a large part in the inference result, and this chemical structure being present in both compounds presumably accounts for this similarity. The presence of this commonality in the chemical structure that strongly influenced the toxicity assessment suggests that this structure is one that warrants attention when assessing toxicity.

In this way, inference factor identification technology was used to indicate which chemical structures influenced the assessment. A comparison of data having similar core tensors also highlighted a chemical structure present in both compounds that played a large part in the inference result. To the extent that these results have a rational interpretation, the inferences generated by Deep Tensor can be considered rational.



a) Compound A

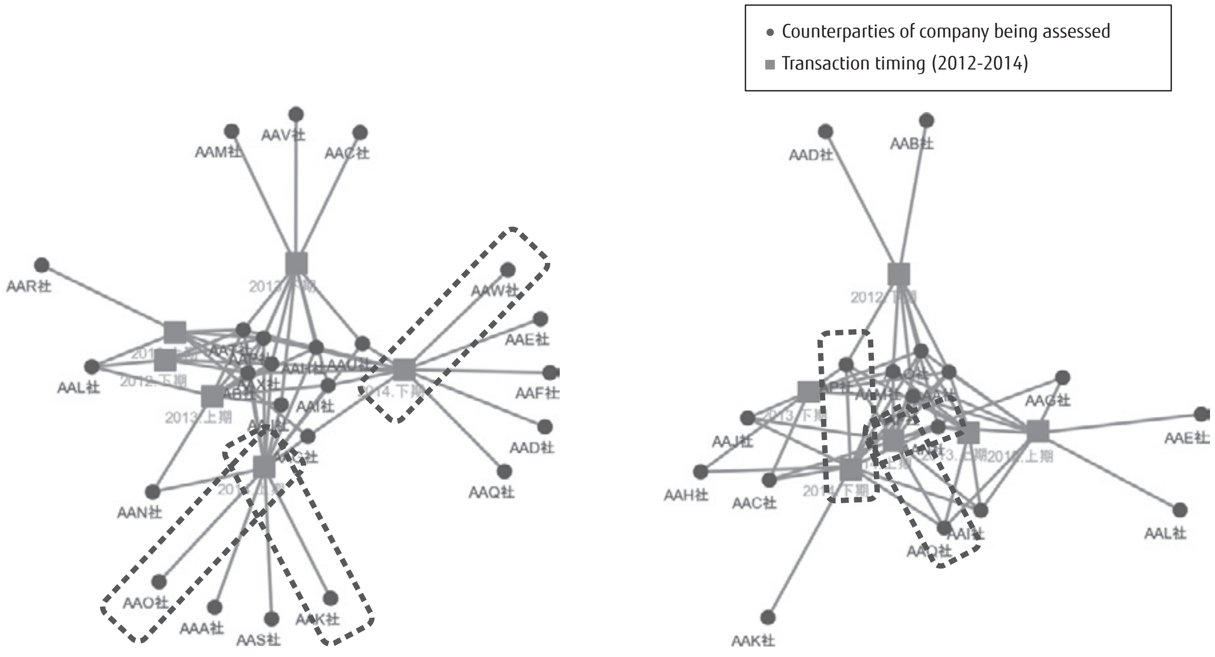
b) Compound B

Figure 4
Use of technique on assessment of toxic chemical compounds.

4.2 Financial application

This example relates to credit assessment at a financial institution, involving the use of records of inter-company transactions held by the financial institution to assess the risk of providing finance to particular companies. This is done by using these records to build a graph of the inter-company transactions involving the company in question and using Deep Tensor to assess, from the features of this graph, whether the risk of financing the company is high or low. The way this is done is fundamentally the same as in the toxicity example above.

Figure 5 shows two transaction graphs. The graphs express the relationships between the companies being assessed (Company A and Company B) and the between transaction counterparties, also taking into account when transactions took place. In this figure, round nodes represent counterparties and square nodes indicate when the transactions occurred. Although it cannot be seen in the figure, the



a) Transaction graph for Company A
(Inference result: High risk)

b) Transaction graph for Company B
(Inference result: Low risk)

Figure 5
Use of technique on transactional relationships graphs.

transaction graph edges also contain information about the monetary value of each transaction.

Although Companies A and B have a similar level of transactions (in terms of number of counterparties and monetary value), their transaction graphs have different characteristics. Company A has many one-time customers, meaning companies with which all transactions occur in only a single time period. Company B in contrast has many repeat customers, meaning companies with transactions across multiple time periods. Based on these characteristics, Deep Tensor assessed Company A as high risk and Company B as low risk.

Those transactional relationships that were identified by inference factor identification technology as making a relatively large contribution to the result are highlighted in Figure 5 by dotted lines. That transactions with one-time customers make a large contribution in the case of Company A indicates concern about transactional relationships that rely on this type of customer. Similarly, that transactions with repeat customers make a large contribution in the case of Company B indicates an acknowledgment that the company has steady transactional relationships with this type of customer.

In this way, use of inference factor identification technology highlighted the transactional relationships that influenced the assessments by Deep Tensor of companies that pose different finance risks despite having transaction graphs that are similar in size. A rational interpretation was obtainable by looking at the characteristics of these transactional relationships.

5. Conclusion

This paper described the inference factor identification technology used to explain inferences made by Deep Tensor and presented examples of its application. The technology works by training a linear regression model to approximate the inference results made by Deep Tensor based on the core tensors. The degree to which each element of the core tensors contributes to the inference results obtained by this training is then transformed into the corresponding degrees of contribution of the graph data, thereby indicating the degree to which each edge in the graph data contributes to the inference results. The technology was used in medical and financial applications to demonstrate that it could indicate which elements of the graph data contributed to a Deep Tensor inference result.

The "explanation" provided by the technology described here comes in the form of information on which elements contributed to an inference, but this is not in itself an adequate explanation. What is needed, rather, to obtain a better explanation is the domain understanding to be able to tie this "explanation" to what it is the data actually represents. It is only by doing so that it becomes possible to encourage new discoveries or take into account law and ethics in decisions. Fujitsu Laboratories has published details of an "explainable AI" that combines inference factor identification technology with knowledge graphs[6], this being one of the ways we intend to address this issue. We intend both to continue developing this "explainable AI" to make it more useful in practice and to utilize it on the FUJITSU Human Centric AI Zinrai platform service for AI.[7]

References

1) Fujitsu : Fujitsu Technology to Elicit New Insights from Graph Data that Expresses Ties between People and Things.
https://www.fujitsu.com/global/about/resources/news/press-releases/2016/1020-01.html

2) K. Maruhashi: "Deep Tensor: Eliciting New Insights from Graph Data that Express Relationships between People and Things." FUJITSU Sci Tech J. Vol. 53, No. 5, pp. 26–31, 2017.
https://www.fujitsu.com/global/documents/about/resources/publications/fstj/archives/vol53-5/paper05.pdf

3) K. Maruhashi et al.: Learning Multi-Way Relations via Tensor Decomposition with Neural Networks. Proc. of Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018), 2018.
https://pdfs.semanticscholar.org/bab9/54548db21034436ee4def167664a1790ba86.pdf

4) M. T. Ribeiro et al.: "'Why Should I Trust You?': Explaining the Predictions of Any Classifier." Proc. of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2016), pp. 1135–1144, 2016.
http://www.kdd.org/kdd2016/papers/files/rfp0573-ribeiroA.pdf

5) U.S. Department of Health & Human Services: Tox21 Data Challenge 2014.
https://tripod.nih.gov/tox21/challenge/data.jsp

6) Fujitsu: Fujitsu Fuses Deep Tensor with Knowledge Graph to Explain Reason and Basis Behind AI–Generated

Findings.
https://www.fujitsu.com/global/about/resources/news/press-releases/2017/0920-02.html

7) Fujitsu: Zinrai AI Business Solutions.
https://www.fujitsu.com/global/solutions/business-technology/ai/

Tatsuru Matsuo
Fujitsu Laboratories Ltd.
Mr. Matsuo is engaged in R&D on the possibilities for explanation in machine learning.

Yusuke Oki
Mr. Oki was engaged in R&D of explainability in machine learning at Fujitsu Laboratories up to March 2020 prior to his retirement.

Koji Maruhashi
Fujitsu Laboratories Ltd.
Dr. Maruhashi is engaged in R&D on the possibilities for explanation in machine learning.