Knowledge graph-based entity importance learning for multi-stream regression on Australian fuel price forecasting

Dennis Chow
Faculty of Engineering and Information Technology
University of Technology Sydney, Australia
Che.K.Chow@student.uts.edu.au

Anjin Liu
Centre for Artificial Intelligence,
Faculty of Engineering and Information Technology
University of Technology Sydney, Australia
Anjin.Liu@uts.edu.au

Guangquan Zhang
Centre for Artificial Intelligence,
Faculty of Engineering and Information Technology
University of Technology Sydney, Australia
Guangquan.Zhang@uts.edu.au

Jie Lu
Centre for Artificial Intelligence,
Faculty of Engineering and Information Technology
University of Technology Sydney, Australia
Jie.Lu@uts.edu.au

***Abstract*—A knowledge graph (KG) represents a collection of interlinked descriptions of entities. It has become a key focus for organising and utilising this type of data for applications. Many graph embedding techniques have been proposed to simplify the manipulation while preserving the inherent structure of the KG. However, scant attention has been given to the investigation of the importance of the entities (the nodes of KGs). In this paper, we propose a novel entities importance learning framework that investigates how to weight the entities and use them as a prior knowledge for solving multi-stream regression problems. The framework consists of KG feature extraction, multi-stream correlation analysis, and entity importance learning. To evaluate the proposed method, we implemented the framework based on Wikidata and applied it to Australian retail fuel price forecasting. The experiment results indicate that the proposed method reduces prediction error, which supports the weighted knowledge graph information as a means for improving machine learning model accuracy.**

***Index Terms*—data stream, regression, knowledge graph**

I. Introduction

In today's setting, where data is available in abundance, ensuring that information is organised and structured is vital. A knowledge graph (KG) is a knowledge base for structured knowledge representation that has an intuitive and interrelated architecture [1]. Structured data is a requirement for applications and can assist in information retrieval, enhanced query accuracy and knowledge integration [2]. KGs have enabled a range of applications, including domain-specific information portals for query answering [3], [4] and knowledge management for the increasing proliferation of information [5], [6]. However, many applications remain unexplored such as KG application in multi-stream regression models.

In the current Big Data era, huge amounts of streaming data are generated by government and industry from multiple sources, known as multiple streams (multi-stream), such as sensors and marketing activities [7]. Many time series and data stream mining techniques have been applied for modelling and solving related problems. However, most of these consider that streaming data from different devices are isolated and have not investigated the correlation between them. In this paper, we focus on utilizing KGs for solving multiple data stream regression problems (multi-stream regression). The intuitive idea is to create a multi-stream network to capture and maintain the interrelationship between streams, where the nodes in the network are the dynamic data streams and the edges are the interrelationships between streams. With a multi-stream network, collaborative stream learning can be achieved.

However, how to learn the interrelationship between streams is still an open problem. Current regression model advancements have involved algorithm improvement [8], [9], and even though algorithm improvements have achieved improved performance, the approach is limited by the data available to the model. KGs provide a contextual understanding of an entity, which is traditionally a missing aspect in the majority of data streams [10]. It is hypothesised that providing the regression model with semantic knowledge will result in enhanced regression model accuracy. In other words, the interrelationship between streams can be acquired by the KG-based stream similarity learning.

The motivation of this study is to exploit KG semantics to improve the accuracy of multi-stream regression rather than only using the correlation retrieved from historical data. In this paper, we focus on multi-stream forecasting which is based on our proposed multi-stream regression model. The main contributions of this paper are:

1) A framework for learning KG entity importance via KG semantic feature extraction and multi-stream correlation analysis.

2) An algorithm for utilising KG information for collaborative multi-stream regression and forecasting.
3) An application based on the proposed framework for Australian retail fuel price forecasting.

To evaluate the performance of the proposed regression model, we apply the algorithm to the task of forecasting Australian fuel prices. The results demonstrate that the proposed algorithm is beneficial to forecasting accuracy.

The rest of this paper is organized as follows. The current advancements in KG, data stream forecasting and regression models are introduced in Section II. In Section III, we propose our KG based multi-stream regression framework. In Section IV, we apply the algorithms to a multi-stream forecasting problem: Australian retail fuel price forecasting. Finally, we present our conclusions and recommendations for future work in Section V.

II. Literature Review

In this section, we formally present the definition of the KG. Then the data stream forecasting problem and the state-of-the-art regression algorithms are discussed.

A. Knowledge graph

A $knowledge\ graph$ (KG) is a data structure that is used to contextualise entities and their relationships. In the KG framework, entities are represented as nodes and the relationship between entities as edges. The structure describes an interconnected web of information that is also highly intuitive. KG can be produced from a variety of corpora including unstructured text such as web articles and semi-structured text. KGs have also been built from existing knowledge bases [11].

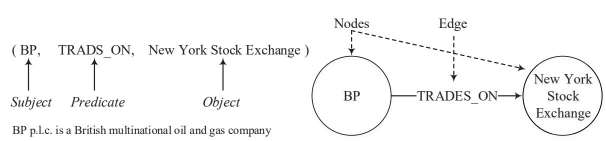

Fig. 1. A example of the basic elements of a knowledge graph. The textual representation of the KG (left) and the graphical representation of the KG (right)

In a typical KG, knowledge is described as relational data represented as triples, i.e. (subject, predicate, object). The subject is the source entity that is joined to an object entity via a specified relationship, the predicate. Fig. 1 illustrates the textual and graphical representation of two entities: BP (BP p.l.c. is a British multinational oil and gas company) and New York Stock Exchange related by the relationship, TRADES_ON. Thus, a KG is fundamentally a collation of knowledge triples.

In recent years, advances in the KG field have been driven by the efforts of projects such as NELL [12], Google KG and WikiData that continue to shape the approach of online data extraction and promote interest in KG research. Ongoing research efforts focus on entity and relation extraction techniques such as OpenIE [13], OntoILPER [14] and DIG [15]. Such techniques present increasingly efficient and accurate methods of extracting useful data for KG.

As research in KG extraction and creation mature, so do the opportunities for KG applications. KG-based applications require domain-specific KGs or data for use. Current applications include knowledge retrieval and search enhancement [16]. However, applications beyond this remain unexplored.

B. Data stream forecasting

Data stream forecasting is closely related to time series analysis and multivariate time series analysis. A time series is a set of observations x_t, each one being recorded at a specific time t. A multivariate time series is considered as a vector-valued (multivariate) time series that has $X_t = \{x_{t1}, \ldots, x_{tk}\}$ at each time t. They not only have serial dependence within each component series x_{ti} but also interdependence between the different component series x_{ti} and x_{tj}, $i \neq j$ [17].

From a data-level perspective, both time series and multivariate time series can be considered as a type of data stream. However, a data stream contains much uncertainty regarding time information t, as in the data stream, the t might be missing and only the order of the data is preserved, or the data received in a time slot might have no order information but the time slot order is available. From the learning task perspective, multivariate time series assumes that for all $x_{ti} \in X_t$ is available and the task is to predict the X_{t+1}, while a data stream considers that some $x_{ti} \in X_t$ could be missing. Therefore, data stream algorithms may need to infer the missing values first and then predict the X_{t+1}.

Time series regression models are widely used for forecasting in business and econometric applications [18], however, research in multi-stream regression is much more limited. Multi-stream data may reveal trends beyond the patterns recognised in a single stream, and thus multi-stream models can outperform single-stream models [19], [20]. As an example, in relation to Australian local fuel prices, the multi-stream view is significant as gas stations encounter local market competition while also being affected by global variables such as exchange rates and stock market indices.

C. Regression analysis

Regression analysis is a statistical process to estimate the relationships among variables and has been widely used for prediction and forecasting [17]. A regression model is a function that describes the relationships between the independent variables X, and the dependent variable y, with some unknown parameters w. denoted as $y = f(X, w)$.

Bayesian regression is a regression model defined in probabilistic terms, with explicit priors on the parameters. The Bayesian probabilistic model is given below, where the output y is assumed to be Gaussian distributed around Xw:

$$p(y|X, \omega, \varepsilon) = \mathcal{N}(y|Xw, \varepsilon),$$

where w is the coefficient vector that $w = (w_1, \ldots, w_d)$, and ε is treated as a random variable that is to be estimated from the data. The priors can have regularising effects, such as $L1$

regularisation (i.e. Lasso) or $L2$ regularisation (i.e. Ridge), which stabilise the predicted values by reducing variance. It is precisely this ability to penalise over-complex models that makes the Bayesian approach highly effective [21].

Decision tree regression is a non-parametric supervised learning method that infers simple decision rules from data for prediction. The model is obtained by recursively partitioning the data space such that the samples with the same labels are collated. The model creates nodes which are partitioned based on an impurity function $H(X)$. For regression, a common criterion is the mean squared error (MSE):

$$\bar{y}_m = \frac{1}{N_m} \sum_{i \in N_m} y_i,$$

$$H(X_m) = \frac{1}{N_m} \sum_{i \in N_m} (y_i - \bar{y}_m)^2,$$

where X_m is the training data in node m, and N_m is the number of data instances in the node. The simplicity of the method means that the speed of predictions is faster than other methods. Despite its simplicity, the predictive performance of decision tree applications is maintained [22].

Support vector regression is an application of support vector machines which employs kernels to solve regression problems. A kernel function is used to map the original space into a higher dimensional space and vice versa. Using kernels allows a model to learn a linear function in the kernel-induced space and map the function back into the original space. Due to the flexibility of this method, it remains a highly researched topic for application and model optimisation [23], [24].

In addition to the above regression models, nearest neighbors regression, perceptron and neural network regression, least absolute shrinkage and selection operator (LASSO) are also well-established regression models. Most of these can be adapted as autoregressive models for handling multivariate time series focusing tasks. However, very few consider semantic information as prior knowledge for variable correlation analysis.

III. KNOWLEDGE GRAPH-BASED MULTI-STREAM REGRESSION FRAMEWORK

In this section, we formally present the proposed KG-based multi-stream regression model. The model includes knowledge graph feature extraction, multi-stream correlation analysis and entity importance learning. The overall framework is shown in Fig. 2. The intuitive idea is to measure the similarity between two streams from both KG semantic similarity (knowledge-level) and data numerical value correlation (data-level), and then use machine learning algorithms to optimize the semantic information (the entities in the KG) so that the KG semantic similarity and the data value correlation can be synchronized.

A. Knowledge graph feature extraction

The first stage is to extract semantic information from the KG to describe the streams. KGs contain repositories of interrelated facts which suggests the quantifiable significance between entities. Leveraging this similarity measurement requires the transformation of a KG into feature space through feature extraction. This stage involves graph representation and graph embedding. The intuitive idea is to consider the stream as an entity in the KG and to use the correlated edges and nodes in the KG to describe the stream entity. For example, the feature in Fig. 1 is 'TRADES_ON New York Stock Exchange'. As previously described, a KG is constructed on knowledge triples, i.e. (subject, predicate, object), thus, a KG feature can be described as a unique predicate object combination. The set of features X_i^{KG} of Stream_i from the KG is the feature space, denoted as

$$X_i^{KG} = \{f_j\}_{(j \in 1,\ldots,n)}, where f_j \in (predicate, object).$$

Accordingly, the features to describe the similarity between stream_1 and stream_2 can be formulated as follows.

$$X_{1,2}^{KG} = \mathrm{SimFeat}(X_1^{KG}, X_2^{KG}). \tag{1}$$

One naive similarity feature function is the indicator function of the union of X_1^{KG} and X_2^{KG}. Since feature exploration is a critical problem for graph-related tasks [25]–[27], how to control the details of the semantic information extracted from KG is still an open problem. In the application section, we discuss how to control semantic information extraction for the case study.

After extracting the features, it is possible to create an $m \times n$ feature matrix, where m is the number of subject entities and n is the number of KG features. In this way, it is possible to merge multiple KG entities into a single feature matrix.

For a given feature matrix, the similarity is measured by considering the intersection of two entities. Conceptually, this is a comparison of the number of adjacent entities between the target entity and the object entity against the total number of entities neighbouring the target entity. The learned similarity for multiple streams is applied as a weighting or threshold for multi-stream regression problems. In this way, better correlated streams have a more significant impact on the target values.

B. Multi-stream correlation analysis

The second stage is to analyse the correlation between multiple streams. For any two streams Stream_1 and Stream_2, their historical data, denoted as S_1, S_2, can be used to estimate the correlation. How to define the correlation to maximize the accuracy of the regression model is a challenging problem. In this section, we apply statistical evaluation to study the strength of a relationship between S_1 and S_2. Statistical analysis is useful to determine if there are possible connections between variables. To allow for the possibly unpredictable change in future observations, it is natural to suppose that each observation s_{1t} is a realized value of a random variable S_{1t} [17]. The unpredictable change in future observations is also known as concept drift [28], [29].

A straightforward method to estimate the correlation between two streaming data S_1 and S_2 is the Pearson correlation

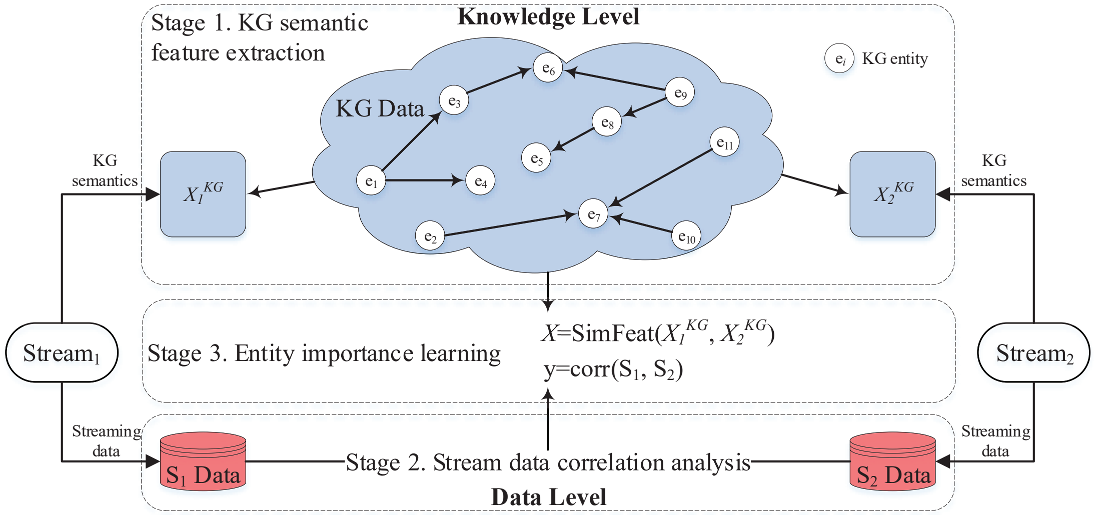

Fig. 2. A KG-based multi-stream regression framework. The fundamental idea of the proposed framework is to utilize knowledge graph data as a prior knowledge to select the most relevant streams for regression. In Stage 1, any two streams, Stream_1 and Stream_2, can be presented as a feature vector based on the KG entities, denoted as X_1^{KG}, X_2^{KG}. Uniting the X_1^{KG}, X_2^{KG}, a new feature vector that presents the similarity between Stream_1 and Stream_2 can be acquired, denoted as $X = \mathrm{SimFeat}(X_1^{KG}, X_2^{KG})$. For Stage 2, the actual data values (data streams S_1, S_2) can be collected from the historical data, and the correlation between S_1 and S_2 is measurable. One can consider the correlation as the target variable, denoted as $y = \mathrm{corr}(S_1, S_2)$. As a result, a learning model can be built to estimate the correlation between two streams from the KG perspective.

coefficient (PCC). In our case, the PCC is calculated as follows,

$$\rho_{S_1,S_2} = \frac{\mathrm{cov}(S_1, S_2)}{\sigma_{S_1}\sigma_{S_2}}, \tag{2}$$

where cov is the covariance and σ is the standard deviation. To remark, the range of PCC is between $+1$ and -1, negative correlation is important as long as we assign it with a proper parameter w.

The intuitive idea of stage 2 is to select the most relevant streams for building the regression model for the target stream. For example, given a multi-stream set $\mathbb{S} = \{S_1, S_2, S_3, S_4\}$, assume the correlations ρ_{S_1,S_2}=0.8, ρ_{S_1,S_3}=-0.6, ρ_{S_1,S_4}=0.05. If we want to build a regression model for S_1, the data from S_2 and S_3 is more useful than S_4. Cross-correlation could also be helpful. Since the main focus of this paper is to embed the data-level correlation into the semantic knowledge, how to select the best correlation function is considered to be future study.

C. Knowledge graph-based entity importance learning

The third stage is KG entity importance learning, that is, acquiring the weights of KG entities via machine learning techniques based on historical data. To make the learning process explainable, we apply linear regression to train the model. The cost function is as follows

$$y_{\rho_{S_1,S_2}} = wX_{1,2}^{KG} + \varepsilon.$$

where $X_{1,2}^{KG}$ is acquired by Eq. (1), $y_{\rho_{S_1,S_2}}$ is the PCC of two streams shown in Eq. (2), w is the parameter vector or the importance of $X_{1,2}^{KG}$, and ε is the error term.

With the entity importance model, for any two given streams, the system can easily calculate their correlation even if the historical data is missing. This model could also be helpful for stream clustering and outlier stream detection.

D. The implementation of the KG-based multi-stream regression and forecasting

This section presents the implementation details of the proposed multi-stream regression and forecasting algorithms. The pseudocodes are shown in Algorithms 1 and 2.

Algorithm 1: KG-based Multi-stream Regression

Input : knowledge graph database, KG
multi-stream training data, $\mathbb{S}$
random training stream size, n
Output: the KG entity importance learning model

Randomly select n streams as sample stream set $\mathbb{S}'$;
for S_i *in* $\mathbb{S}'$ **do**
  for S_j *in* $\mathbb{S}'$ **do**
    if $i \neq j$ **then**
      Extract the features from KG, $X_{i,j}^{KG}$;
      Merge $X_{i,j}^{KG}$ as a matrix, denoted as $M_{X^{KG}}$;
      Compute the PCC $y_{\rho_{S_i,S_j}}$;
      Merge $y_{\rho_{S_i,S_j}}$ as a vector, denoted as y_ρ;
Optimize the learning model $y_\rho = wM_{X^{KG}} + \varepsilon$;
return the learning model

To start with, the system needs a KG database for semantic feature extraction, such as Freebase, DBpedia, YAGO, and NELL as summarized in [1]. The historical data of the streams is also required for data-level correlation analysis. Since the number of available data streams could be very large, a parameter n should be given to control how many streams are used for building the regression model. From line 1 to 4, the system randomly selects a subset of the multi-stream and for looping to compare all pairs of streams to initialize the $y_{\rho_{S_i,S_j}}$ and $X_{1,2}^{KG}$ for building the entity importance learning model. KG semantic feature extraction is implemented in lines 5,6. The data-level correlation analysis is in lines 7,8, and the learning stage is in line 9. Line 10 returns the learning result.

Algorithm 2: KG-based Multi-stream Forecasting

Input : knowledge graph database, KG
KG entity importance learning model, $\mathcal{L}_{KG}$
regression model for forecasting, $\mathcal{L}_{reg}$
stream relevance threshold, default $\alpha = 0.5$
lag selection range, default $\beta \in \mathbb{Z}_{\leq 10}^{+}$

Output: the forecasting result of the stream $\hat{y}$

1 **for** S_i *in* $\mathbb{S}$ **do**
2   Extract the features from KG, $X_{i,tar}^{KG}$;
3   Estimate the correlation $\hat{y}_{\rho_{S_i,S_{tar}}} = \mathcal{L}_{KG}(X_{i,tar}^{KG})$;
4   **if** $|\hat{y}_{\rho_{S_i,S_{tar}}}| \geq \alpha$ **then**
5   Store the data of S_i as training data for regression, denoted as M_S^{train};
6 **Initialize** $\text{RMSE}_{\max} = 0$;
7 **for** $\beta \in \mathbb{Z}_{\leq 10}^{+}$ **do**
8   Evaluate the $\hat{\text{RMSE}}$ of $\mathcal{L}_{reg}$ on M_S^{train} with lag β;
9   **if** $\text{RMSE}_{\max} < \hat{\text{RMSE}}$ **then**
10         $\text{RMSE}_{\max} = \hat{\text{RMSE}}$;
11         $\beta_{\max} = \beta$;
12 Forecast, $\hat{y} = \mathcal{L}_{reg}(M_S^{\text{test}}, \beta_{\max})$;
13 **return** $\hat{y}$

Denote the target stream for forecasting as Stream_{tar}. For each stream, we estimate the semantic similarity via the KG with the learned entity importance ($\mathcal{L}_{KG}$) and store the strong relevant streams for regression in lines 1-5. The α is a hyperparameter and we consider the streams with estimated correlation greater than 0.5 as relevant streams. As a rule of thumb, for the absolute value of PCC: 0.0-0.3: very weak; 0.30-0.5: weak; 0.5-0.7: moderate; 0.7-0.9: strong; 0.9-1.0: very strong [30]. Admittedly, calculating the PCC of S_i, S_{tar} from the data-level to select strong relevant streams is more accurate. However, as discussed in subsection II-B, data streams require preprocessing which is a complicated task. If the system can find the most relevant streams based on the knowledge-level information, it could save the cost of data stream preprocessing. From lines 6 to 11, we select the best performing lag parameter based on a grid search with a range $\beta \in \mathbb{Z}_{\leq 10}^{+}$. The best lag value $\beta_{\max}$ with the build regression model $\mathcal{L}_{reg}$ will be applied for forecasting in line 12. The forecasting result is returned in line 13.

IV. Application and Experiment

This section presents an implementation of the proposed framework in Section III. The chosen application is fuel price forecasting for Australian gas retail stations. In Australia, fuel is an essential resource for the continued expansion and development of cities and remains a highly integrated commodity that powers industries, businesses and day-to-day lives. For this reason, the effects of the global production and pricing of oil have significant impacts on the Australian economy [31], [32], as well as motorists' daily activities. It follows that fuel price is a highly speculated resource due to its significant effect on many economies.

The objective of this application is to analyse historical petroleum prices and predict the price for the next seven days. In Australia, fuel stations are in close proximity to one another, hence factors such as convenience and price are pivotal in this highly competitive market. Motorists choose convenience and pricing as major factors in deciding which fuel station to go to fill up their cars. Motorists can benefit from accurate fuel price prediction, such as purchasing petroleum at the lowest point in a price cycle. The evaluation metric is the root mean square error (RMSE) of the predictions and the actual values. According to the prediction results, it is envisioned that the proposed KG-based multi-stream regression can improve the accuracy of fuel price forecasts by utilising the contextual knowledge between local entities. The original fuel price dataset is available online[1].

A. Dataset description

Wikidata KG dataset. Two domain-specific KGs are generated for feature extraction and entity learning. The first KG knowledge base relates to the business domain to identify relationships between business entities. The second KG knowledge base relates to the geographic domain to determine the relationships between suburb entities. The KGs were created by extracting knowledge triples from the Wikidata knowledge base from filtered entity sets. Filtered sets were produced from the SPARQL query service to ensure the relevance of the extracted data and are summarised in Table I. KG extraction was implemented using the Python programming language and purpose-built library, pywikibot, for data extraction from Wikidata. For the predicates in the KGs, a custom defined schema was used based on the Wikidata property. For example, for the Wikidata property 'instance of', the predicate used was IS_A. The predicate schema is summarised in Table II. Finally, the feature matrix was extracted from each KG. The KG statistics are summarised in Table III.

Using the feature matrices extracted from the KGs, similarity was correlated for 250 gas stations. The size of the resulting correlation training matrix was 31,375 by 271, i.e. (gas station combinations) by (number of features). The correlation training matrix was implemented in the regression model by imposing a threshold correlation value of $\alpha = 0.5$ to data streams. For regression modelling, fuel prices were gathered

[1]https://data.nsw.gov.au/data/dataset/fuel-check

for 1,580 gas stations twice daily over a 2-year period from August 2016 to July 2018. The gas station dataset also includes information such as address, suburb, postcode, brand and fuel types. In this paper, the fuel type predicted was P98.

TABLE I
FILTERED SET CRITERIA (WIKIDATA QUERY SPARQL)

KG domain	Set size	SPARQL filter criteria
Business	400	Instance of (P31) business (Q4830453) Industry (P452) petroleum industry (Q862571)
Geographic	600	Instance of (P31) suburb (Q188509), and Located in the administrative territorial entity (P131) New South Wales (Q3224).

TABLE II
PREDICATE SCHEMA FOR KG. EXAMPLES (PREDICATE OBJECT COMBINATION)

Wikidata property	predicate	Example
instance of (P31)	IS_A	IS_A business
industry (P452)	IN_INDUSTRY	IN_INDUSTRY petroleum industry
follows (P155)	WAS_PREVIOUSLY	WAS_PREVIOUSLY Enterprise Oil
owned by (P127)	OWNED_BY	OWNED_BY BlackRock
owner of (P1830)	OWNER_OF	OWNER_OF Showa Shell Sekiyu
country (P17)	IS_LOCATED_IN	IS_LOCATED_IN Australia
population (P1082)	HAS_POPULATION	HAS_POPULATION 7111
postal code (P281)	HAS_POSTCODE	HAS_POSTCODE 2007

TABLE III
KG STATISTICS

KG domain	No. Entities	No. Relationship	No. Features
Business	592	1401	227
Geographic	891	1847	44

Australian fuel price dataset. For prediction, 8 gas stations were randomly chosen, as summarised in Table IV. Due to the absence of available data for some petrol stations in August, each data stream was truncated to match the length of the shortest data stream. The data was truncated to ensure sets were comparable and all data points were genuine and would not produce misleading results. After pre-processing, each data stream had 1254 data points equating to 628 days of actual data.

B. Forecasting parameter settings

The regression models, Bayesian ridge regression (BRR) and decision tree regression (DTR) were implemented for evaluation. Each model was initially trained with the 8 selected stations with an initial window of 360 data points (i.e. 6 months) for a prediction period of 14 units ahead (i.e. 7 days). Models were evaluated over 10 regressive lags for stream S_1 to determine the lag which minimised the prediction error. In

TABLE IV
SELECTED GAS STATIONS STREAMS

Stream ID	Gas station	Suburb	Fuel type
S_1	7-Eleven	Adamstown	P98
S_2	7-Eleven	Albion Park Rail	P98
S_3	7-Eleven	Argenton	P98
S_4	BP Connect	Caringbah	P98
S_5	BP Connect	Mortdale	P98
S_6	Coles Express	Alexandria	P98
S_7	Coles Express	Ultimo	P98
S_8	Coles Express	Waterloo	P98

Fig. 3, the mean square error (MSE) is plotted against each lag.

The lag which minimised MSE was used to perform sliding window prediction over the following 730 data points (i.e. 1 year). Regression models were implemented using the Python programming language and machine learning library scikit-learn. The model parameters are summarised in Table V.

TABLE V
REGRESSION MODEL PARAMETERS FOR SCIKIT-LEARN PYTHON PACKAGE

Model	Function	Parameters
BRR	linear_model.BayesianRidge	α_1=1e-06, α_2=1e-06, λ_1=1e-06, λ_2=1e-06
DTR	tree.DecisionTreeRegressor	criterion=mse, max_depth=10

C. Forecasting results

To quantitatively evaluate the performance of the proposed multi-stream regression model for forecasting tasks, the forecast error is measured by the difference between the actual value and the forecast value for the corresponding period: $E_t = Y_t - F_t$ where E is the forecast error at period t, Y is the actual value at period t, and F is the forecast for period t. The root mean square error is calculated by $\text{RMSE} = \sqrt{\frac{\sum_{t+1}^{N} E_t^2}{N}}$.

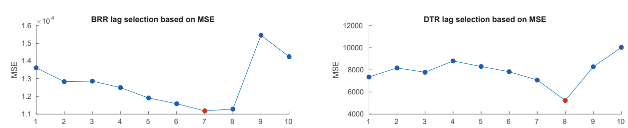

Fig. 3. Lag selection for BRR and DTR

Fig. 4 shows the prediction results for the three regression models, BRR, DTR and the collaborative multi-stream regression models KG-BRR, KG-DTR. From the results, we observe that there is a large discrepancy between the predicted results between regression models. The reason for this is that simple regression models such as DTR are more readily operational than parameter heavy models such as BRR. The dependency on parameters is further exemplified from the MSE trend for the BBR and DTR lag selection plot in Fig. 3. From Table VI we observe that collaborative multi-stream regression methods outperform baseline regression models. This indicates that entity similarity learned from the KG is

TABLE VI

RMSE SUMMARY OF EVALUATION RESULTS. SINGLE-BRR AND SINGLE-DTR ARE THE FORECASTING RESULTS ONLY BASED ON THE TARGET STREAM. BRR AND DTR ARE THE FORECASTING RESULTS BASED ON MULTI-STREAMS WITHOUT CONSIDERING THE RELEVANCE BETWEEN STREAMS. KG-BRR AND KG-DTR ARE THE FORECASTING RESULTS BASED ON THE KG ESTIMATED CORRELATION.

	Single-BRR	BRR	KG-BRR	Single-DTR	DTR	KG-DTR
S_1	221.48	228.02	228.79	163.90	167.37	174.26
S_2	312.60	282.06	287.7	274.42	215.44	192.97
S_3	183.59	188.76	189.56	180.22	171.39	160.17
S_4	302.74	275.14	270.65	259.21	171.74	176.83
S_5	298.31	285.74	282.38	297.43	181.73	190.4
S_6	279.78	253.63	250.05	230.76	153.87	155.3
S_7	301.99	268.41	267.3	277.93	196.83	196.15
S_8	281.97	252.01	247.25	233.90	161.95	167.59
Aver. rmse	272.81	254.22	**252.96**	239.72	177.54	**176.71**

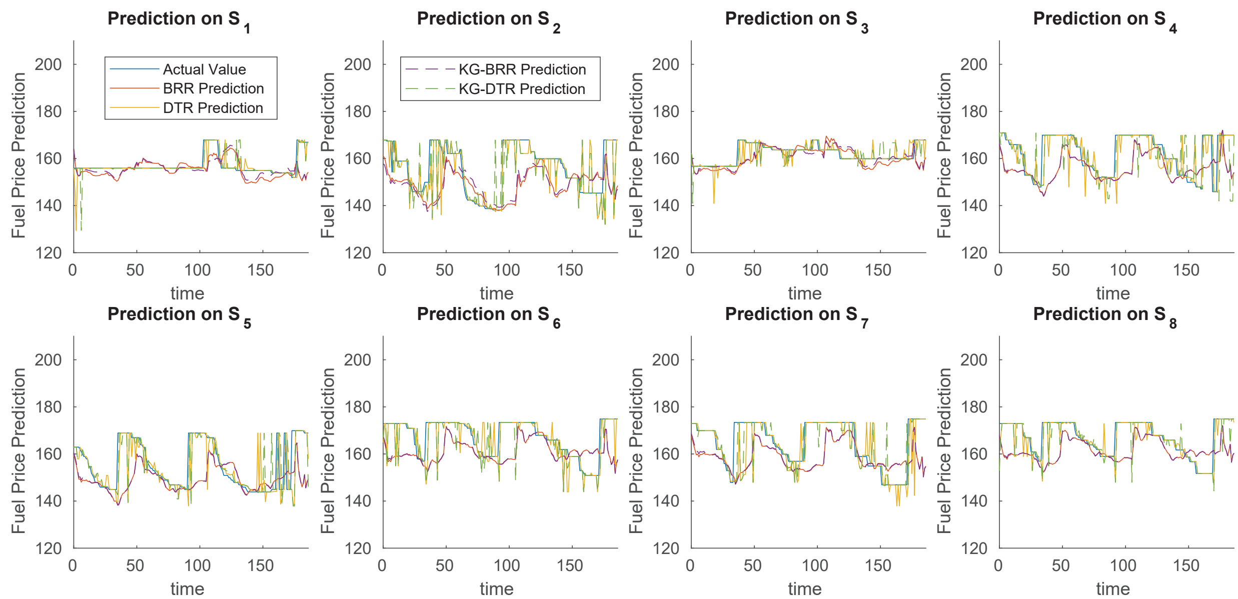

Fig. 4. Prediction plots for 8 gas stations using BRR, DTR, KG-BRR and KG-DTR prediction models for Feb, Mar, Apr 2018. Parameters: $t_{\text{ahead}} = 14$, $\text{BRR}_{\text{lag}} = 7$, $\text{DTR}_{\text{lag}} = 8$

useful and improves regression accuracy. It is manifest that multi-stream forecasting is more accurate than single stream. However, the improvement due to using KG is marginal. This is reasonable because we only use the selected eight streams to build the regression model. The KG estimated correlation only detected one or two streams as not relevant. This problem could be resolved by increasing the stream pool. In other words, the results could be further improved by involving more streams, and this will be our next step in multi-stream related research.

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed a framework for collaborative multi-stream regression using correlations learned from KGs. The framework detailed KG feature extraction, multi-stream correlation analysis, entity importance learning and was applied to predict fuel prices in Australian gas stations from real-world datasets. In the experiments, multi-stream forecasting was evaluated against existing regression models BRR and DTR. The experiment results show that the proposed algorithms improved the performance, which indicates the potential of the framework to improve multi-stream regression accuracy.

The following research directions are suggested for future work: (1) The current implementation only considered a threshold condition from the learned similarity. A weighting method for sample selection could be explored to enhance the results. (2) The current implementation work of the correlation information is applied to data that is external to the regression models. Exploring an integration of the KG into the regression algorithm itself would enhance the results and allow the approach to be more accessible. (3) We verify the effectiveness of the method for three regression models only. The framework can be extended to more sophisticated regression models (e.g. ARIMA) to further demonstrate the applicability of the model. (4) The RMSE results were marginal for some streams. Thus, undertaking further optimisation of regression models is suggested to consolidate the results.

ACKNOWLEDGMENT

The work presented in this paper was supported by the Australian Research Council (ARC) under Discovery Project DP190101733. We acknowledge the support of NVIDIA Corporation with the donation of GPU used for this research.

REFERENCES

[1] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 12, pp. 2724–2743, 2017.

[2] A. Schmidt, J. Hoffart, D. Milchevski, and G. Weikum, "Context-sensitive auto-completion for searching with entities and categories," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 2911461: ACM, 2016, pp. 1097–1100.

[3] P. Ernst, A. Siu, and G. Weikum, "Knowlife: a versatile approach for constructing a large knowledge graph for biomedical sciences," *BMC Bioinformatics*, vol. 16, no. 1, p. 157, 2015.

[4] P. Chen, Y. Lu, V. W. Zheng, X. Chen, and B. Yang, "Knowedu: A system to construct knowledge graph for education," *IEEE Access*, vol. 6, pp. 31 553–31 563, 2018.

[5] M. Galkin, S. Auer, M.-E. Vidal, and S. Scerri, "Enterprise knowledge graphs: A semantic approach for knowledge management in the next generation of enterprise information systems," in *Proceedings of the 19th International Conference on Enterprise Information Systems*, 2017, pp. 88–98.

[6] S. Auer, V. Kovtun, M. Prinz, A. Kasprzik, M. Stocker, and M.-E. Vidal, "Towards a knowledge graph for science," in *Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics*. ACM, 2018, pp. 1–6.

[7] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," *IEEE Transactions on Knowledge and Data Engineering*, 2018.

[8] E. Faria, A. Ponce de Leon Ferreira Carvalho, and J. Gama, "Minas: multiclass learning algorithm for novelty detection in data streams," *Data Mining and Knowledge Discovery*, vol. 30, no. 3, pp. 640–680, 2016.

[9] A. Osojnik, P. Panov, and S. Džeroski, "Multi-label classification via multi-target regression on data streams," *Machine Learning*, vol. 106, no. 6, pp. 745–770, 2017.

[10] R. Sousa and J. Gama, "Multi-label classification from high-speed data streams with adaptive model rules and random rules," *Progress in Artificial Intelligence*, vol. 7, no. 3, pp. 177–187, 2018.

[11] A. Ismayilov, D. Kontokostas, S. Auer, J. Lehmann, and S. Hellmann, "Wikidata through the eyes of dbpedia," *Semantic Web*, vol. 9, no. 4, pp. 493–503, 2018.

[12] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, and B. Kisiel, "Never-ending learning," *Communications of the ACM*, vol. 61, no. 5, pp. 103–115, 2018.

[13] J. L. Martinez-Rodriguez, I. Lopez-Arevalo, and A. B. Rios-Alvarado, "Openie-based approach for knowledge graph construction from text," *Expert Systems with Applications*, vol. 113, pp. 339–355, 2018.

[14] R. Lima, B. Espinasse, and F. Freitas, "Ontoilper: an ontology- and inductive logic programming-based system to extract entities and relations from text," *Knowledge and Information Systems*, vol. 56, no. 1, pp. 223–255, 2018.

[15] P. Szekely, C. A. Knoblock, J. Slepicka, A. Philpot, A. Singh, C. Yin, D. Kapoor, P. Natarajan, D. Marcu, and K. Knight, "Building and using a knowledge graph to combat human trafficking," in *Proceedings of the 2015 International Semantic Web Conference*. Springer, 2015, pp. 205–221.

[16] L. Shao, Y. Duan, X. Sun, H. Gao, D. Zhu, and W. Miao, "Answering who/when, what, how, why through constructing data graph, information graph, knowledge graph and wisdom graph," in *Proceedings of the 2017 International Conference on Software Engineering and Knowledge Engineering*, 2017, pp. 1–6.

[17] P. J. Brockwell, R. A. Davis, and M. V. Calder, *Introduction to time series and forecasting*, 3rd ed. Springer, 2016, vol. 2.

[18] L. Wu, Y. Wang, and S. Pan, "Exploiting attribute correlations: A novel trace lasso-based weakly supervised dictionary learning method," *IEEE transactions on cybernetics*, vol. 47, no. 12, pp. 4497–4508, 2017.

[19] Y. Hwang, A. Tong, and J. Choi, "Automatic construction of non-parametric relational regression models for multiple time series," in *Proceedings of the 2016 International Conference on Machine Learning*, 2016, pp. 3030–3039.

[20] M. Jin and C. J. Spanos, "Brief: Bayesian regression of infinite expert forecasters for single and multiple time series prediction," in *Proceedings of the 54th IEEE Conference on Decision and Control (CDC)*. IEEE, 2015, pp. 78–83.

[21] D. J. MacKay, "Bayesian interpolation," *Neural computation*, vol. 4, no. 3, pp. 415–447, 1992.

[22] E. V. Wolputte, E. Korneva, and H. Blockeel, "Mercs: Multi-directional ensembles of regression and classification trees," in *Proceedings of the Thirth-Second AAAI Conference on Artificial Intelligence*, 2018.

[23] H.-Y. Cho, H.-J. Kim, and Y.-H. Kim, "Parameter space analysis of genetic algorithm using support vector regression," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. 3208770: ACM, 2018, pp. 35–36.

[24] G. Ou, Y. Wang, L. Huang, W. Pang, and G. M. Coghill, "ε-distance weighted support vector regression," in *Proceedings of the 22nd Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2018, pp. 209–220.

[25] S. Pan, J. Wu, X. Zhu, G. Long, and C. Zhang, "Task sensitive feature exploration and learning for multitask graph classification," *IEEE transactions on cybernetics*, vol. 47, no. 3, pp. 744–758, 2017.

[26] S. Pan, R. Hu, G. Long, J. Jiang, L. Yao, and C. Zhang, "Adversarially regularized graph autoencoder for graph embedding," in *Proceedings of the 27th International Joint Conferences on Artificial Intelligence*, 2018, pp. 2609–2615.

[27] S. Pan, J. Wu, X. Zhuy, C. Zhang, and P. S. Yuz, "Joint structure feature exploration and regularization for multi-task graph classification," in *Proceedings of the 32nd IEEE International Conference on Data Mining*. IEEE, 2016, pp. 1474–1475.

[28] A. Liu, J. Lu, F. Liu, and G. Zhang, "Accumulating regional density dissimilarity for concept drift detection in data streams," *Pattern Recognition*, vol. 76, pp. 256–272, 2018.

[29] A. Liu, Y. Song, G. Zhang, and J. Lu, "Regional concept drift detection and density synchronized drift adaptation," in *Proceedings of the Twenty-sixth International Joint Conference on Artificial Intelligence*, 2017, pp. 2280–2286.

[30] D. E. Hinkle, W. Wiersma, and S. G. Jurs, *Applied statistics for the behavioral sciences, the 5th ed.* Boston: Houghton Mifflin, 2003.

[31] K. Drachal, "Comparison between bayesian and information-theoretic model averaging: Fossil fuels prices example," *Energy Economics*, vol. 74, pp. 208–251, 2018.

[32] J. Wang, G. Athanasopoulos, R. J. Hyndman, and S. Wang, "Crude oil price forecasting based on internet concern using an extreme learning machine," *International Journal of Forecasting*, vol. 34, no. 4, pp. 665–677, 2018.