# A Services Classification Method Based on Heterogeneous Information Networks and Generative Adversarial Networks

Xiang Xie, Hunan University of Science and Technology, China

Jianxun Liu, Hunan University of Science and Technology, China*

Buqing Cao, Hunan University of Science and Technology, China*

Mi Peng, Hunan University of Science and Technology, China

Guosheng Kang, Hunan University of Science and Technology, China

Yiping Wen, Hunan University of Science and Technology, China

Kenneth K. Fletcher, University of Massachusetts, Boston, USA

## ABSTRACT

With the rapid development of service computing and software technologies, it is necessary to correctly and efficiently classify web services to promote their discovery and application. The existing service classification methods based on heterogeneous information networks (HIN) achieve better classification performance. However, such methods use negative sampling to randomly select nodes and do not learn the underlying distribution to obtain a robust representation of the nodes. This paper proposes a web services classification method based on HIN and generative adversarial networks (GAN) named SC-GAN. The authors first construct a HIN using the structural relationships between web services and their attribute information. After obtaining the feature embedding of the services based on meta-path random walks, a relationship-aware GAN model is input for adversarial training to obtain high-quality negative samples for optimizing the embedding. Experimental results on real datasets show that SC-GAN improves classification accuracy significantly over state-of-the-art methods.

## KEYWORDS

*Generative Adversarial Networks*, *Heterogeneous Information Network*, *Relationship-Aware*, *Services Classification*, *Web Services*

## INTRODUCTION

Web services are gradually becoming the mainstream technology for implementing service-oriented architecture (SOA) applications. With the emergence of many more SOA-based applications, more and more Web services are available on the Internet today. For this reason, the rapid and accurate discovery and selection of required Web services have become a fundamental challenge in service

*Corresponding Author

computing. In addition, the lack of a formal description model, too little description text, and the irregular description language, further increases the challenge of Web services discovery and selection (Ye, Cao et al. 2019).

Web services classification techniques have been studied and proposed by many researchers previously. The main goal is to reduce the space and time required for Web services search to improve the efficiency and quality of Web service discovery. Most of these studies classify Web services based on their functional attributes (Wang, Yang et al. 2017) (Xia, Fan et al. 2014). They typically employ TF-IDF, cosine similarity, and other similarity measures to determine the functional similarity between Web services based on Web services description language (WSDL) documents (Xia, Fan et al. 2014). Furthermore, several researchers have used LDA topic models or their extensions (Shi, Liu et al. 2019) (Shi, Liu et al. 2017) (Cao, Liu et al. 2017a) (Cao, Liu et al. 2019) to mine hidden topic information in Web services. These topic-model-based works represent Web services using low-dimensional topic vector features and classify Web services by computing similarities based on these topic vectors. However, it is very challenging for these approaches that only consider service content information to achieve good results because of the short length and sparse nature of WSDL documents (Cao, Liu et al. 2016).

Web services are also directly or indirectly related to other information (e.g., Tags, Mashups, etc.), which characterize the functional properties of a Web service from several perspectives(Cao, Liu et al. 2017b). Therefore, several methodologies exist to classify Web services using auxiliary relationships such as tags. Although these methodologies improve the accuracy of Web service classification to a certain extent, they rely on attribute information such as textual description information and labels that do not fully consider the complex structural interactions between Web services (combination and shared labeling relationships).

Several objects link Web services to form a natural heterogeneous information network and provide new ideas for some special Web service classification situations. This has recently led to several researchers focusing on studying node representation learning for heterogeneous information networks (HIN)(Shi, Li et al. 2016) and have applied it for service classification. HIN aims to learn to map input spaces to lower-dimensional spaces while preserving heterogeneous structure and semantics, one of the most promising of such works are Metapath2vec(Dong, Chawla et al. 2017), HERec(Shi, Hu et al. 2018) and Hin2vec(Fu, Lee et al. 2017). Though these methods achieve significant classification accuracy improvement in node classification, they also have limitations. First, they usually sample negatively by randomly selecting existing nodes in the network. Second, they focus on capturing rich semantic information over heterogeneous information networks without paying attention to the underlying node distribution. For these reasons, they do not perform well under real network situations, which tend to be sparse and noisy.

To tackle these challenges, some researchers have adopted generative adversarial network (GAN) models11 to learn potential robust representations for various applications(Ding, Tang et al. 2018) (Yu, Zhang et al. 2017). GAN depends on the idea of adversarial learning, where the discriminator and the generator compete with each other in order to train a better discriminatory model and learn the underlying data distribution. Early adoption of this technique has shown some promising preliminary results on GAN-based network representation learning(Pan, Hu et al. 2018) (Wang et al. 2018) (Fu, Lee et al. 2017). In particular, (Hu, Fang et al. 2019) proposed an adversarial learning approach based on heterogeneous information networks and showed better performance in node classification tasks.

Inspired by the above research, we propose a hybrid Web service classification method based on HIN and GAN called SC-GAN. The proposed method merges heterogeneous information networks and generative adversarial network techniques. Our proposed method uses a heterogeneous information representation model that fully extracts the structural features of Web services and adopts GAN training to capture a more robust representations of Web service nodes. Consequently, the proposed method improves the accuracy of Web service classification. To sum up, the main contributions of this paper are as follows:

- We extend adversarial learning to Web service node representation for heterogeneous information networks, which preserves the rich semantics of Web service heterogeneous information networks, ensures the robustness of the learned service node representations, and subsequently improves the accuracy of service classification.
- We propose a novel service classification method by fusing heterogeneous information networks and adversarial training, which captures the relationships for rich semantics and provides an efficient mechanism for generating negative samples.
- We perform experiments based on real datasets from ProgrammableWeb, and the experimental results validate the effectiveness of the proposed method, which improves the accuracy and quality of Web service classification.

The rest of the paper is organized as follows: Section 2 provides background information to our proposed work. We describe our proposed SC-GAN method in Section 3 and report the experimental evaluation and analysis in Section 4. In Section 5, we present some current works in literature. Finally, we conclude and provide an outlook on the future work of this paper in Section 6.

## PRELIMINARY

In this section, we introduce concepts related to heterogeneous information networks and generative adversarial networks (GAN) that we use in this paper.

*Definition 2.1*. **Heterogeneous information network**. A heterogeneous information network (HIN) is a form of a directed graph $\mathcal{G} = \left\{ S, E, \mathcal{A}, \mathcal{R}, \phi, \varphi \right\}$, in which $S$ denotes the set of nodes, $E$ presents the interaction behaviors between different types of objects. $\phi$ is the object type mapping function such that $\phi : S \to \mathcal{A}$ for node types and $\varphi$ is the link mapping function such that $\varphi : E \to \mathcal{R}$, for relation types, in which $\mathcal{A}$ and $\mathcal{R}$ denote the set of node and relation types, respectively and while $|\mathcal{A}| + |\mathcal{R}| > 2$.

*Definition 2.2*. **Meta-path**. A meta-path is the path that defines the connection between objects on the network schema, formally represented as $\mathcal{A}_1 \xrightarrow{\mathcal{R}_1} \mathcal{A}_2 \xrightarrow{\mathcal{R}_2} \cdots \xrightarrow{\mathcal{R}_l} \mathcal{A}_{l+1}$, and complicated relation between object types on the path as $\mathcal{R} = \mathcal{R}_1 \circ \mathcal{R}_2 \circ \cdots \circ \mathcal{R}_l$. Among them, Where $\circ$ represents the composite operator between relations, $\mathcal{A}_i$ denotes the object type and $\mathcal{R}_i$ denotes the relation type.

*Definition 2.3*. **Adversarial Generative Networks (GAN).** GANs are regarded as a max-mini game among two players, namely the generator $G$ and the discriminator $D$. The specific optimization form is as follows:

$$\min_{\theta^G} \max_{\theta^D} E_{x \sim P_{data}} \left[ \log D\left( x; \theta^D \right) \right] + E_{x \sim P_Z} \left[ \log\left( 1 - D\left( G\left( z; \theta^G \right); \theta^D \right) \right) \right] \tag{1}$$

The generator $G$ attempts to generate pseudo-samples as similar as possible to the real data using the noise $Z$, from the predefined distribution, where the parameters of the generator are represented. Conversely, the discriminator $D$ intends to discriminate between real data from distribution and pseudo-data from the generator, where $\theta^D$ represents the parameters of the discriminator. It has been found in practice that GAN generally works better if the generator minimizes $\log D\left( \left( G\left( \cdot; \theta^G \right); \theta^D \right) \right)$ instead of $\log\left( 1 - D\left( G\left( \cdot; \theta^G \right); \theta^D \right) \right)$ (Reed, Akata et al. 2016).

## SC-GAN METHOD FOR WEB SERVICE CLASSIFICATION

### The Overall Framework

In this section, we describe the proposed SC-GAN method in detail. First, we obtain information such as service description documents, tags, and labels from the Web services repository. We use this information to construct a service heterogeneous information network. Next, we specify the relationship between Web service nodes under the meta-path into the Metapath2vec model. Then we input the obtained service node embedding into a GAN model with a multi-layer perceptron for generative adversarial training to capture a robust representation of the Web service nodes. Finally, we use multivariate logistic regression to predict the category of Web services. The overall framework of our proposed SC-GAN method is shown in Figure 1 and comprises three main stages: (1) service heterogeneous information network construction and Metapath2vec network representation; (2) SC-GAN adversarial training; and (3) Web service classification.

### Service Heterogeneous Information Network Construction and Metapath2vec Network Representation

HIN comprise several types of objects and rich interaction relations. Figure 2 shows a hypothetical example of a Web service HIN. It is made up of a variety of node types (such as Web services, mashups, tags, and description documents) and their semantic relation (such as Mashup-Web service call, Web services shared tags, and Web services containing description documents). Due to their heterogeneous nature, HIN often has extremely rich and complex semantics. Therefore, we first consider the problem of constructing meta-paths to perform relationship mining. We use nodes of the same type to construct symmetrical meta-paths (i.e., the nodes at the start and end of the meta-path are of the same type). Table 1 shows examples of the constructed meta-paths. We combine a priori knowledge of the services network to conclude that service nodes with the same label and subject term may belong to the same category, as well as mashups that invoke the same services may invoke the same category of services.

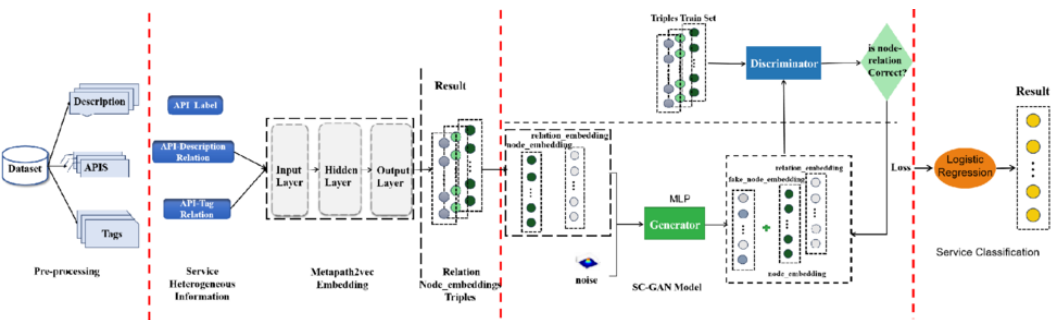**Figure 1.**
**The overall framework of the SC-GAN model**



**Table 1.**
**Meta paths related to services**

| meta-path | semantics |
|---|---|
| SMSMS | Mashups that have called the same service may call services of the same category |
| STS | Services with the same label may belong to the same category |
| SDS | Services with the same subject heading may belong to the same category |

**Figure 2.**
**Service Heterogeneous Information Network**
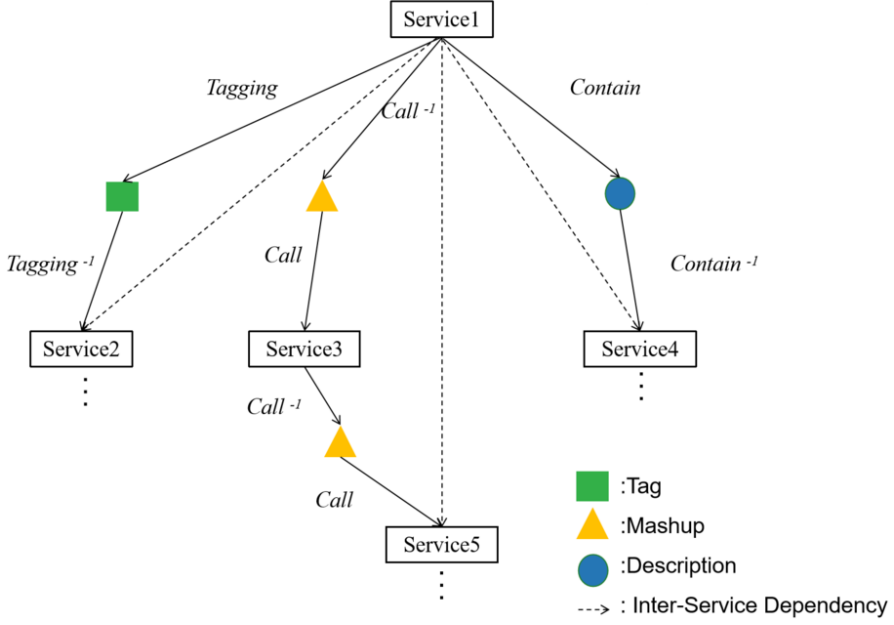


In general, we construct meta-paths via a priori knowledge of the network. After obtaining the meta-path, we learn an embedding of the Web services nodes. The goal of HIN embedding is to learn a mapping function that projects each Web service node into a low-dimensional vector space. By investigating the embedding of heterogeneous graphs, we select the Metapath2vec model for the HIN embedding. Metapath2vec learns network representations through random walking and heterogeneous Skip-gram models based on meta-paths that capture semantic and structural correlations between different types of nodes. The walk path is generated according to the following distribution:

$$
P(n_{k+1} = x \mid n_k = s) = \begin{cases} \dfrac{1}{\left| \mathcal{N}_\rho^1 (s) \right|}, & (s,x) \in E \, and \, s, x \in \mathcal{A}_k, \mathcal{A}_{k+1} \\ 0, & otherwise. \end{cases}
\tag{2}
$$

Where $n_k$ is the $k$-th node in the walk, $\mathcal{N}_\rho^1 (s)$ denotes the set of fist-order neighbors of the meta-path $\rho$. A walk will repeatedly follow the pattern of meta-paths until it reaches a predefined length.

We define a walking path as a meta-path that expresses the possibility that services with the same label belong to the same class. For instance, as shown in Figure 2 and Table 1, "STS" is a meta-path where S and T denote the Web service node type and the Web services label node type, respectively. By employing a Meta-path-based random walking we randomly determine the node type in the next-hop service node using the current and corresponding service node type with the meta-path pattern. Ultimately, meta path-based random wandering strategies capture connections between different node types by ensuring that they are properly fused into the skip-gram model with semantic relationships between them.

## GAN ADVERSARIAL TRAINING

The SC-GAN stage consists of two main competing components, namely discriminator and generator. When a Web service node is given, the generator tries to generate a pseudo-sample associated with it and provides the sample to the discriminator to tell if it is a real or fake sample. The discriminator optimizes its parameters to separate the pseudo-sample from the real sample associated with the given node. This process continues iteratively until a well-trained discriminator forces the generator to produce better pseudo-samples, and the discriminator enhances its judgment criteria. In such an iterative process,both the generator and the discriminator are positively enhanced.

Considering the representation learning that preserves the rich semantics of Web service nodes, the proposed SC-GAN model introduces discriminators and generators with relationship awareness to distinguish various types of semantic relationships between service nodes. As shown in Figure 3, we feed a meta-path based on a service HIN into the GAN for adversarial training. The goal is to train the discriminator such that given a Web service node label and a relation such as tag/tagged, the discriminator can distinguish between any given two services. To generate better samples, we use a generalized generator to generate fake samples such as node service', where service' may not belong to the HIN. The service may be an "average" of service 1, service 3, more similar to the real sample service 2 rather than a fake sample of service 3.

## RELATIONSHIP-AWARE GENERATOR

To tackle the problem of processing multi-relational data, (Bordes, Usunier et al. 2013) proposed the TransE algorithm for distributed vector representation based on entities and relationships. where the head entity embedding plus the relationship embedding of each triplet instance (head, relationship, tail) will equal the tail entity embedding. In the TransE embedding method for heterogeneous information networks, for each triple $\langle s, r, v \rangle$ in the service heterogeneous information network (where $s$ and $v$ denote two service nodes linked by the relation $r$, respectively.), Also, we denote their sets as $s$ and $v \in S$ and $r \in E$ respectively.

**Figure 3.**
**Example of ATA input SC-GAN adversarial training**
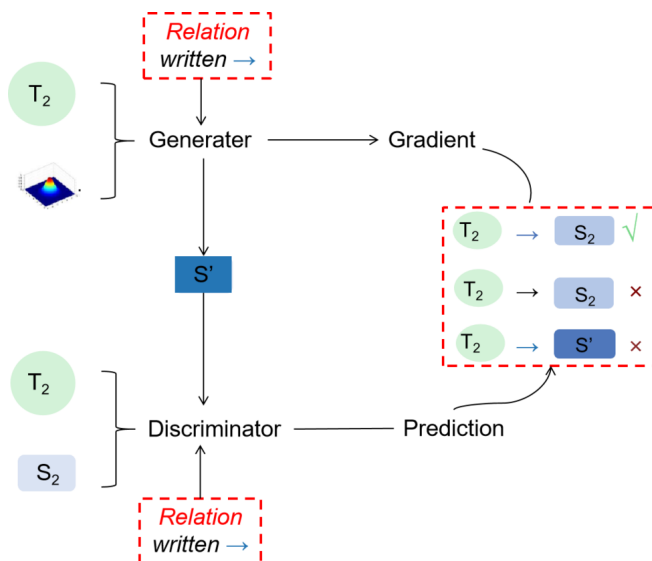
The goal of generator $G$ is to generate pseudo samples to simulate the real samples. For one thing, G perceives relations as well as the discriminator does. Therefore, when we are given a service node $s \in S$ and a relation $r \in E$, the generator aims to generate a pseudo service node $v$ that connected to service node $s$ by relation relation $r$. That is, node $v$ should be as close to the real node as possible, e.g., $\langle s, r, v \rangle \sim P_{\mathcal{G}}$. For the other thing, the generator is generative, which means that the pseudo service node $v$ may be potential rather than discovered in $S$. To satisfy these requirements, the generator uses a matrix specific to the relationship (for relationship awareness) and in a continuous distribution from the underlying layer. Specifically, we use the following Gaussian distribution $N\left(e_s^{G^T} M_r^G, ó^2 I\right)$ for generating samples from the underlying continuous distribution. where $e_s^G \in \mathbb{R}^{d \times 1}$ and $M_r^G \in \mathbb{R}^{d \times d}$ represent the service node embedding of generator $s \in S$ and the relation matrix of $r \in E$, respectively. In other words, for some choice of $ó$, it is a Gaussian distribution. Intuitively, the mean $e_s^{G^T} M_r^G$ represents fake-service nodes that may be connected to $s$ via relation $r$, and the covariance $ó^2 I \in \mathbb{R}^{d \times d}$ denotes potential bias. A simple solution is to generate samples directly from $N\left(e_s^{G^T} M_r^G, ó^2 I\right)$. However, since neural networks have demonstrated powerful capabilities in modelling complex structures(Wan, Li et al. 2015), multi-layer perceptron (MLP) are integrated into generators to enhance the output of pseudo-samples. Therefore, the formula for the generator is as follows:

$$G\left(s, r; \theta^G\right) = f\left(W_L \cdots f\left(W_1 e + b_1\right) + b_L\right) \tag{3}$$

In the above equation, we obtain $e$ from the distribution $N\left(e_s^{G^T} M_r^G, \sigma^2 I\right)$, where $f$ is the activation function, and $W_*$ and $b_*$ denote the weight matrix and deviation vector for each layer, respectively.

The generator wants to generate samples that obey the true data distribution as close as possible to deceive the discriminator:

$$\mathcal{L}^G = \mathbb{E}_{s, r \sim P_{\mathcal{G}}, e_v' \sim G\left(s, r; \theta^G\right)} - log D\left(e_v' \mid s, r\right) + \lambda^G \theta^{G2}_{\ 2} \tag{4}$$

where $ë^G > 0$ controls the regularization term. We can optimize the model by minimizing the loss function of the generator G.

## RELATIONSHIP-AWARE DISCRIMINATOR

A distinction must be made on the HIN between real and fake service nodes for a given relationship. Therefore, the relationship-aware discriminator $D\left(e_v \mid s, r; è^D\right)$ evaluates the connectivity between the service node pairs $s$ and $v$ for the relationship. In particular, $s \in \mathcal{V}$ is the given service node, $r \in E$ denotes the given relation from the HIN of service, $e_v$ represents the embedding of the sample service node (which may be either true or false), and $è^D$ is the $D$ model parameters. Essentially, the probability that $D$ outputs a sample connection to a service under relation B can be quantified as follows:

$$D\left(e_v \mid s, r; \theta^D\right) = \frac{1}{1 + exp\left(-e_s^{D^T} M_r^D e_v\right)} \tag{5}$$

Here $e_s^D \in \mathbb{R}^{d \times 1}$ is the embedding of the learnable service node $s \in \mathcal{V}$, $e_v$ is the input embedding of sample $v$, and $M_r^D \in \mathbb{R}^{d \times d}$ is a learnable relationship matrix for relation $r \in E$. $\grave{e}^D = \left\{ e_s^D : s \in S, M_r^D : r \in E \right\}$ is the model parameters of the discriminator D, That is, we combine all node embedding and the relation matrix to learn the discriminator parameter.

If $v$ is a positive sample associated with $s$ through the relation $r$, the probability should be high, but when it is a negative sample, the probability should be low. Usually, the sample $v$ forms a triple $\langle s, r, v \rangle$ with a given service node $s$ and relation $r$, and each triad falls into one of the following three cases in terms of polarity. Inspired by the conditional GAN (Hu, Fang, et al. 2019), each case has an impact on the loss of a part of the discriminator.

**Case 1:** The connection relationship is correct. Such triples are considered to be positive examples if nodes $s$ and $v$ are indeed connected by the correct relation r on the HIN graph, e.g., <Service 1, Label 1, tagging>, as shown in Figure 3. and can be modeled by the following loss.

$$\mathcal{L}_1^D = \mathbb{E}_{s,v,r \sim P_{\mathcal{G}}} - logD\left( e_s^D \mid s, r \right) \tag{6}$$

**Case 2:** The connection relationship is incorrect. If the connectivity relation between $s$ and sample $v$ in the HIN is incorrect, e.g. <service1, label1, unlabeled>. The discriminator is also expected to mark them as negative samples because their connectivity does not match the given relation $r \neq r'$. We calculate this part of the loss using the following equation.

$$\mathcal{L}_2^D = \mathbb{E}_{s,v \sim P_{\mathcal{G}}, r' \sim P_{\mathcal{R}'}} - log\left( 1 - D\left( e_v^D \mid s, r' \right) \right) \tag{7}$$

**Case 3:** False service nodes from a relationship-aware generator. If the service node $v$ is a fake node generated by the generator, such as <service 1, label', tagged> in Figure 3. Although it is generated by the generator simulation connected to $s$ under the correct relation $r$. Also, the discriminator recognizes this triple as negative, and this part of the loss can be expressed as follows.

$$\mathcal{L}_3^D = \mathbb{E}_{s,r \sim P_{\mathcal{G}}, e_v' \sim G\left( s, r; \theta^G \right)} - log\left( 1 - D\left( e_v' \mid s, r \right) \right) \tag{8}$$

Where the embedding $e_v'$ of the pseudo-sample $v$ is extracted from the learned distribution of the generator G.

It is not to be overlooked that there may be a fourth case where triples are pseudo-pairs from relation-independent generators, however, such negative triples are easier than negative numbers in case 2 or case 3 distinguished from positive values. Therefore, although our model has the flexibility to incorporate the fourth case, they are not considered here. Therefore, we integrate the above three components to train the discriminator.

$$\mathcal{L}^D = \mathcal{L}_1^D + \mathcal{L}_2^D + \mathcal{L}_3^D + \lambda^D \theta^{D2}_2 \tag{9}$$

To avoid over-fitting we use $\lambda^D > 0$ to control the regular term and optimize the parameter $\theta^D$ of the discriminator by minimizing $\mathcal{L}^D$.

## Service Classification

The service node embedding representations of the SC-GAN model are input into a fully connected layer, as well as the probability distribution of all candidate web service categories using a multivariate logistic regression function. Logistic regression is used to predict a model with the probability of different possible outcomes of a dependent variable with category distribution and converts the output value of multi-category into relative probabilities, indicating the service node belonging to a particular category. The method to compute this is shown in equation (10). Among them, The $P$ is the probability that the model will predict which category the training data belongs to respectively, $k$ denotes the number of candidate Web service categories, $w$ is the weight matrix corresponding to the service node, and $e_{s_i}^D$ denotes the feature embedding of the vector.

$$P\left(k \mid e_{s_i}^D\right) = \frac{exp\left(w_k e_{s_i}^D\right)}{\sum_{j=1}^{N} exp\left(w_j e_{s_j}^D\right)} \tag{10}$$

In the model training process, for all K possible classification outcomes, K-1 independent binary logistic regression models are run for all K possible classification outcomes, with one of the categories considered as the primary category during the process, and then the rest of the K-1 categories and the chosen primary category are regressed separately. Taking the maximum value in log-likelihood is equivalent to taking the minimum value of the loss function, where the log-likelihood is calculated as shown in Equation (11).

$$log\left(Loss\right) = \sum_{i=1}^{N} \left\{ -w_i e_{s_i}^D + log \sum_{j=1}^{K} exp\left(w_j e_{s_j}^D\right) \right\} \tag{11}$$

## EXPERIMENTAL EVALUATION AND RESULT ANALYSIS

### Data Set Pre-Processing and Experimental Setup

To evaluate our proposed model we crawled from ProgrammableWeb.com 17,782 Web API and related information. This information includes their names, description documents, primary and secondary Categories, and tags. We use this data in several experiments to evaluate and validate our proposed SC-GAN for Web services classification.

To improve the classification accuracy, we first preprocessed the data by tokenizing, removing stop words, and stemming. Then, we select the top 10, 15, 20, 25, and 30 categories as our experimental datasets. Among them, The top 30 categories with the largest number of words were distributed as shown in Table 2. In the experiment, we use a train-test split of 80/20 where we choose 80% of the data as the training set and 20% of the data as the test set. The number of random walks is 10, the walk step size is 20, the representation dimension size d is 128, and the window size c is 5. In addition, to ensure the objectivity of the experiment, the common parameters between SC-GAN and other methods are set to the same value, while the remaining parameters are set to the default optimal values.

## BASELINE METHODS

To evaluate and validate the effectiveness of our proposed model, we compare it with some strong baselines on the Web services classification tasks. We chose two homogeneous network representation

**Table 2.**
**Top 30 categories by number**

| Category | Number | Category | Number | Category | Number |
|---|---|---|---|---|---|
| Tools | 850 | Telephony | 338 | Games | 240 |
| Financial | 758 | Reference | 308 | Photos | 228 |
| Messaging | 601 | Security | 305 | Music | 221 |
| eCommerce | 546 | Search | 301 | Stocks | 200 |
| Payments | 526 | Email | 291 | Cloud | 195 |
| Social | 501 | Video | 289 | Data | 187 |
| Enterprise | 472 | Travel | 284 | Bitcoin | 173 |
| Mapping | 437 | Education | 275 | Other | 165 |
| Government | 369 | Transportation | 259 | Project Management | 165 |
| Science | 368 | Advertising | 254 | Weather | 164 |

methods, i.e., Node2vec, LINE, and two heterogeneous network representation methods, i.e., Metapath2vec, HERec. Here we give a little information on the baseline works.

- *__Node2vec(Grover and Leskove 2016):__ Node2vec is similar to Deep-walk, which improves the random wandering strategy by defining two hyper-parameters p and q to control the weighting of depth-first and breadth-first strategies. This combination of DFS neighborhood and BFS neighborhood for graph embedding allows the generated random walks to reflect both depth-first and breadth-first sampling, thus improving the effectiveness of the network embedding.*
- *__LINE(Tang, Qu et al. 2015):__ By training the first and second order proximity between vertices, both global and local network structure information is obtained, and edge sampling methods are used to measure the closeness between nodes and obtain similar nodes for classification.*
- *__Metapath2vec(Dong, Chawla et al. 2017):__ A network representation learning method specifically for heterogeneous graphs. The meta-path-based random wandering strategy ensures the different types of node semantic relationships are appropriately incorporated into the skip-gram model, which is then used to predict the local domain vertex information for each vertex and thus learn the feature representation for each vertex.*
- *__HERec(Shi, Hu et al. 2018):__ A Meta-path-based random wandering method was designed to generate many meaningful sequences of nodes in order to maintain semantic HIN embedding, thus fusing the embedding for application to downstream recommendation tasks.*

## EXPERIMENTAL PERFORMANCE

In this section, we compare the classification performance of all approaches and analyze the experimental results. We selected the top 10, 15, 20, 25, and 30 service categories in the datasets for our experiments, and used Macro and Micro F1 values as performance evaluation metrics. The results of the node classification are reported in Table 3,and the graphs of the changes in Micro F1 and Macro F1 with the number of categories are shown in Figure 4 and 5. We made the following observations:

- *The classification performance decreases as the number of categories taken from the datasets increases. The reasons for this may be: 1) the more categories in the datasets, the more information (noise) and complexity the data contains, which makes the model more difficult to classify. 2) The*

**Table 3.**
**Comparison of Experimental Results of Different Models**

| Model | Number of categories | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 10 | | 15 | | 20 | | 25 | | 30 | |
| | Micro_ F1 | Macro_ F1 | Micro_ F1 | Macro_ F1 | Micro_ F1 | Macro_ F1 | Micro_ F1 | Macro_ F1 | Micro_ F1 | Macro_ F1 |
| LINE | 0.5396 | 0.5276 | 0.4788 | 0.4444 | 0.4601 | 0.433 | 0.4352 | 0.3986 | 0.4297 | 0.3884 |
| Node2vec | 0.5534 | 0.5377 | 0.4817 | 0.4494 | 0.4631 | 0.428 | 0.4336 | 0.3997 | 0.4257 | 0.3882 |
| HERec | 0.6056 | 0.5914 | 0.543 | 0.5082 | 0.4843 | 0.4423 | 0.4462 | 0.4145 | 0.4324 | 0.3818 |
| Metapath2vec | 0.6103 | 0.6042 | 0.5446 | 0.5106 | 0.4997 | 0.4663 | 0.4864 | 0.4541 | 0.4438 | 0.4028 |
| SC-GAN | **0.6643** | **0.656** | **0.5521** | **0.5291** | **0.507** | **0.4729** | **0.4929** | **0.4579** | **0.4562** | **0.4226** |

**Figure 4.**
**Changes in Micro F1 with the Number of Categories**
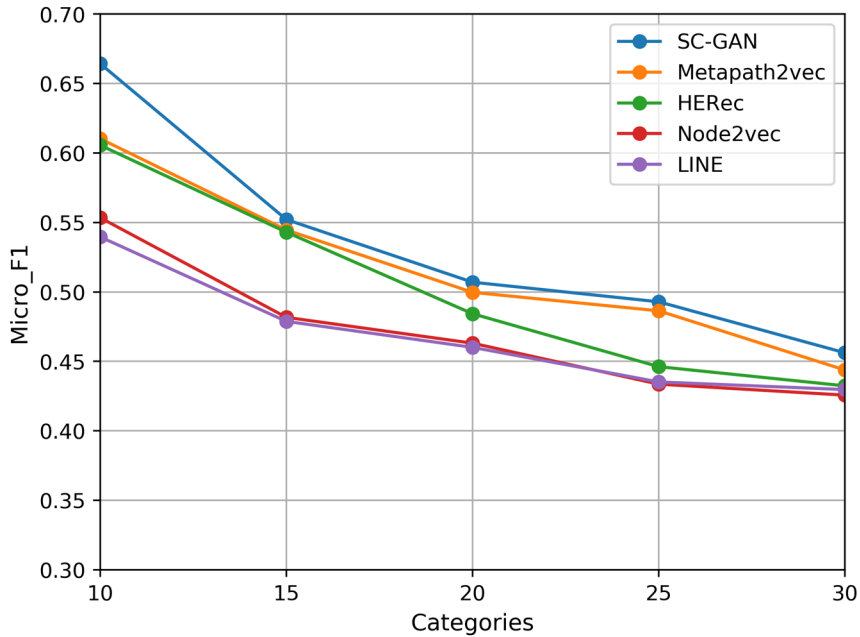


categories are ranked based on the number of Web API included in each category, and lower-ranked categories use fewer API and have less useful information available, thus affecting the effectiveness of the categories.

- *Among the two kinds of baselines, most heterogeneous information network-based methods (HERec, Metapath2vec) outperform the traditional homogeneous information network methods (LINE, Node2vec) in most cases. An intuitive explanation is that those methods based on heterogeneity can better capture the rich higher-order structural information on HIN, i.e. HERec and Metapath2vec embed heterogeneous information in the form of meta-paths into the node representation vector. It should be noted that our model SC-GAN based on meta path-guided neighborhoods can jointly consider first-order neighborhood information with higher-order semantic messages.*

**Figure 5.**
**Changes in Macro F1 with the Number of Categories**
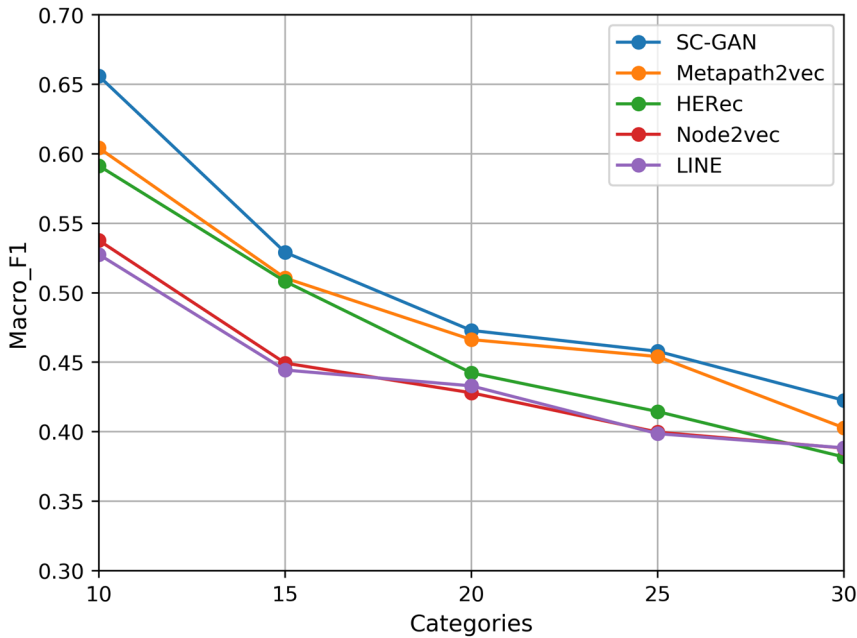


- *In all cases, the proposed SC-GAN model achieves the best performance, with a maximum relative improvement of 8.8% compared to the best baseline Metapath2vec. The results validate the effectiveness of SC-GAN on the task of predicting service node classification, which shows that the more accurately SC-GAN learns to retain semantic representations more robustly through the adversarial principle, the more accurate the representation of service node feature vectors tends to be, and the more effective the classification will be.*

## PARAMETER ANALYSIS

In this section, we investigate the sensitivity of the datasets to the parameters of concern. More results are presented in Figure 6. The learning curves of the generator (Gan) and discriminator (Dis) of the SC-GAN, which we used as an example for 25 service categories, are shown in Figure 6. From the figure, it can be seen that (1) after the initial fluctuation of loss occurs, Gan and Dis start to play the max-min game to gradually reduce the loss of both, And after approximately 12-time units of training against each other, the losses converge and the winner, i.e., Dis achieves better performance. (2) to more accurately evaluate the classification performance of the SC-GAN model, the service node representation of the discriminator optimized by adversarial training is finally selected, and Micro_F1 reaches its maximum after about 12-time units, and the classification quality decreases when more time units are trained due to over-fitting.

## RELATED WORKS

Web service discovery and mining have become a popular topic in recent years with the development of service computing. Some studies prove that efficient Web service classification can effectively

**Figure 6.**
**Learning Curve of SC-GAN on the Datasets: (a) Loss change (b) Classification performance change**
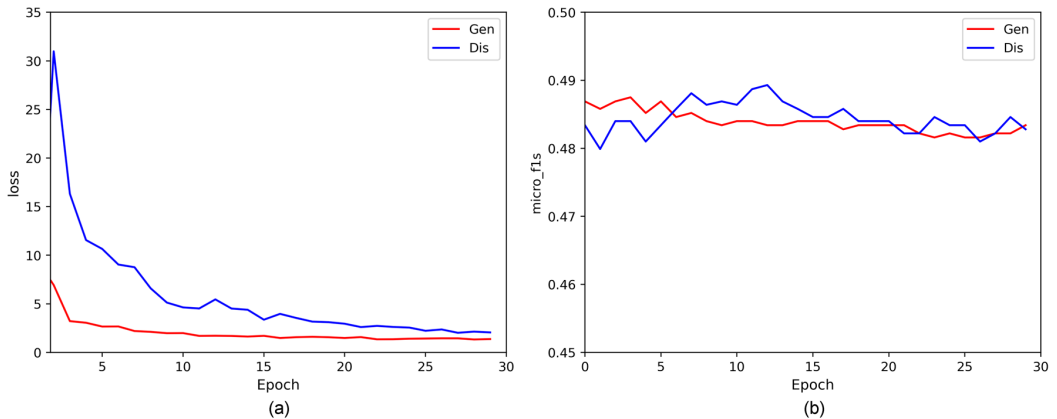


improve the performance of Web service discovery(Liu, C. 2011) (Wang, Li et al. 2020). At present, numerous studies on Web service classification are mainly based on functional semantics. Among them, (Crosso, Zunino et al. 2008) proposed to segment the elements in WSDL to remove deactivated words, group them into word roots and then classify them using different classification algorithms. (Katakis, Tsoumakas et al. 2009) addressed the problem of automatic classification of Web services in their application domain by considering textual description and semantic annotation of Web services. Considering the dynamic nature of service networks, some researchers have introduced time sequences for incremental learning. (Ye, Cao et al. 2019) combined all discrete features in description documents of Web services to predict Web service categories using Wide&Bi-LSTM model. (Chen, Cao et al. 2019) designed the topic attention mechanism-enhanced mobile application classification method by using the LSA model for global topic modelling of mobile application content text and then modeled local hidden representation of content text by the BiLSTM model. (Shi, Liu et al. 2018) presented a probabilistic topic model MR-LDA by considering multiple Web service relationships, which can model the relationship between Web services combined and shared tags between Web services. Considering that the training samples are limited and the classification performance of the tail category is much worse than that of the head category, (Liu, Li et al. 2022) proposed a multi-information fusion-based classification method for few-sample Web services by using the knowledge learned from the head category and the information contained in the category names to improve the classification of the tail category. However, it is difficult for the topic model to obtain effective service representations due to the sparsity of WSDL document data, and its effectiveness of service classification is not significantly improved. (Cao, Peng et al. 2022) used Web API combinations and shared annotation relationships to construct Web API relationship networks, and then applied a self-attention mechanism to calculate the attention coefficients of different neighbouring nodes in the Web API relationship networks. The high-quality representation results are combined with multidimensional service quality attributes for service recommendation.

Recently, several researchers have begun to study network embedding models and techniques used in classification tasks. Network embedding (Cui, Wang et al. 2018) can learn node representations of preserved structures, where typical models and methods include random walks (e.g. Node2vec(Grover and Leskovec 2016)), neighborhood representations (e.g. LINE(Tang, Qu et al. 2015)) etc. However, most of these models and methods are only applicable to homogeneous networks but are unavailable for learning to preserve the rich semantic representations in HIN. To combine the advantages of HIN and network embedding, some methods(Dong, Chawla et al. 2017) (Shi, Hu et al. 2018) (Wang, Sun et

al. 2019) (Reed, Akata et al. 2016) have been used for representation learning in HIN, leveraging meta-path-based embedding with context-preserving semantics to achieve better results in classification tasks. Among them, (Wan, Li et al. 2015) proposed that it is possible to achieve very effective active learning and good classification performance in HIN using class-level meta-paths. (Wang, Ji et al. 2019) first uses node-level attention to aggregate nodes connected by the same meta-path and utilizes semantic-level attention to fuse information from different meta-paths. (Du, Guo et al. 2019) presented to achieve classification by weighted active learning of multi-semantic meta-paths to effectively alleviate the problem of data sparsity.

These previous works focus on capturing rich semantic information over heterogeneous information networks, but do not pay attention to the underlying distribution of nodes and hence lack robustness to the often sparse and noisy real networks. According to our investigation, generative adversarial networks (GAN)(Wang, Gou et al. 2017) utilize adversarial principles to learn more robust representations and have shown excellent performance in many problems(Ding,Tang et al. 2018) (Yu, Zhang et al. 2017). Generally, they impose a fixed prior distribution on the embedding space to enhance the robustness of the learned representation(Pan, Hu et al. 2018) (Wang, Wang et al. 2018). However, these approaches ignore the heterogeneity of nodes and relations which fails to capture the rich semantics of HIN. In this context, we propose a Web service classification method that fuses heterogeneous information networks and generative adversarial networks, inspired by the adversarial learning on heterogeneous information networks(Hu, Fang et al. 2019). It utilizes heterogeneous information to preserve the richer semantics of service nodes and adversarial training to capture more robust representations of service nodes, respectively.

## CONCLUSION AND FUTURE WORK

In this paper, we introduce the problem of early summarization and propose a Web service classification method that fuse heterogeneous information networks and generative adversarial networks, i. e. SC-GAN, to address this problem. We first construct a service heterogeneous information network using the structural relationships between Web services and their attribute information respectively. Then we use Metapath2vec for training to obtain low-dimensional service node embedding of services fused with heterogeneous information. Finally, we input the node embedding of services into the adversarial generative network GAN model for adversarial training to obtain higher quality negative samples and a more robust service node representation for Web service classification. Extensive experimental results have demonstrated the superiority of our model in terms of effectiveness and interpretability in service classification.

In future work, a promising direction is to extend the interaction and aggregation module between the service node API and Mashup, and to consider more fine-grained aggregation of meta-paths for better service classification and recommendation.

## ACKNOWLEDGMENT

# REFERENCES

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. *Advances in Neural Information Processing Systems*, 26.

Cao, B., Liu, J., Wen, Y., Li, H., Xiao, Q., & Chen, J. (2019). QoS-aware service recommendation based on relational topic model and factorization machines for IoT Mashup applications. *Journal of Parallel and Distributed Computing*, *132*, 177–189. doi:10.1016/j.jpdc.2018.04.002

Cao, B., Liu, X., Li, B., Liu, J., Tang, M., Zhang, T., & Shi, M. (2016). Mashup service clustering based on an integration of service content and network via exploiting a two-level topic model. *2016 IEEE International Conference on Web Services (ICWS)*, 212-219. doi:10.1109/ICWS.2016.35

Cao, B., Liu, X. F., Liu, J., & Tang, M. (2017a). Domain-aware Mashup service clustering based on LDA topic model from multiple data sources. *Information and Software Technology*, *90*, 40–54. doi:10.1016/j.infsof.2017.05.001

Cao, B., Liu, X. F., Rahman, M. M., Li, B., Liu, J., & Tang, M. (2017b). Integrated content and network-based service clustering and web apis recommendation for mashup development. *IEEE Transactions on Services Computing*, *13*(1), 99–113. doi:10.1109/TSC.2017.2686390

Cao, B., Peng, M., Qing, Y., Liu, J., Kang, G., Li, B., & Fletcher, K. K. (2022). Web API recommendation via combining graph attention representation and deep factorization machines quality prediction. *Concurrency and Computation*, *34*(21).

Chen, J., Cao, B., Cao, Y., Liu, J., Hu, R., & Wen, Y. (2019). A mobile application classification method with enhanced topic attention mechanism. *CCF Conference on Computer Supported Cooperative Work and Social Computing*, 683-695. doi:10.1007/978-981-15-1377-0_53

Crasso, M., Zunino, A., & Campo, M. (2008). Awsc: An approach to web service classification based on machine learning techniques. *Inteligencia Artificial*, *12*(37), 25–36. doi:10.4114/ia.v12i37.955

Cui, P., Wang, X., Pei, J., & Zhu, W. (2018). A survey on network embedding. *IEEE Transactions on Knowledge and Data Engineering*, *31*(5), 833–852. doi:10.1109/TKDE.2018.2849727

Dai, Q., Li, Q., Tang, J., & Wang, D. (2018). Adversarial network embedding. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1). Advance online publication. doi:10.1609/aaai.v32i1.11865

Ding, M., Tang, J., & Zhang, J. (2018). Semi-supervised learning on graphs with generative adversarial nets. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 913-922. doi:10.1145/3269206.3271768

Dong, Y., Chawla, N. V., & Swami, A. (2017). metapath2vec: Scalable representation learning for heterogeneous networks. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 135-144. doi:10.1145/3097983.3098036

Du, Y., Guo, W., Liu, J., & Yao, C. (2019). Classification by multi-semantic meta path and active weight learning in heterogeneous information networks. *Expert Systems with Applications*, *123*, 227–236. doi:10.1016/j.eswa.2019.01.044

Fu, T. Y., Lee, W. C., & Lei, Z. (2017). Hin2vec: Explore meta-paths in heterogeneous information networks for representation learning. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 1797-1806.

Grover, A., & Leskovec, J. (2016). node2vec: Scalable feature learning for networks. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 855-864. doi:10.1145/2939672.2939754

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural collaborative filtering. *Proceedings of the 26th International Conference on World Wide Web*, 173-182. doi:10.1145/3038912.3052569

Hu, B., Fang, Y., & Shi, C. (2019). Adversarial learning on heterogeneous information networks. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 120-129. doi:10.1145/3292500.3330970

Katakis, I., Meditskos, G., Tsoumakas, G., & Bassiliades, N. (2009). On the combination of textual and semantic descriptions for automated semantic web service classification. *IFIP International Conference on Artificial Intelligence Applications and Innovations*, 95-104. doi:10.1007/978-1-4419-0221-4_13

Liu, C. (2011). *Research on a feature vector-based web service discovery algorithm*. Jilin University.

Liu, Y., Li, B., Wang, J., Li, D., & Ma, Y. (2022). Multi-information fusion based few-shot Web service classification. *Future Generation Computer Systems*, *130*, 231–240. doi:10.1016/j.future.2021.12.020

Min, S., Jianxun, L., Dong, Z., Buqing, C., & Yiy, W. (2018). Web service clustering method based on multiple relational topic model. *Chinese Journal of Computers*, *41*(66), 1–16.

Pan, S., Hu, R., Long, G., Jiang, J., Yao, L., & Zhang, C. (2018). Adversarially regularized graph autoencoder for graph embedding. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 2609-2615. doi:10.24963/ijcai.2018/362

Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., & Lee, H. (2016). Generative adversarial text to image synthesis. *International Conference on Machine Learning*, 1060-1069.

Shi, C., Hu, B., Zhao, W. X., & Philip, S. Y. (2018). Heterogeneous information network embedding for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *31*(2), 357–370. doi:10.1109/TKDE.2018.2833443

Shi, C., Li, Y., Zhang, J., Sun, Y., & Philip, S. Y. (2016). A survey of heterogeneous information network analysis. *IEEE Transactions on Knowledge and Data Engineering*, *29*(1), 17–37. doi:10.1109/TKDE.2016.2598561

Shi, M., Liu, J., Zhou, D., Tang, M., & Cao, B. (2017). WE-LDA: a word embeddings augmented LDA model for web services clustering. *2017 IEEE International Conference on Web Services (ICWS)*, 9-16. doi:10.1109/ICWS.2017.9

Shi, M., Liu, J. X., Zhou, D., Cao, B. Q., & Wen, Y. P. (2019). Multi-relational topic model-based approach for web services clustering. *Chinese Journal of Computers*, *42*(4), 820–836.

Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015). Line: Large-scale information network embedding. *Proceedings of the 24th International Conference on World Wide Web*, 1067-1077. doi:10.1145/2736277.2741093

Wan, C., Li, X., Kao, B., Yu, X., Gu, Q., Cheung, D., & Han, J. (2015). Classification with active learning and meta-paths in heterogeneous information networks. *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 443-452. doi:10.1145/2806416.2806507

Wang, H., Li, J., Yu, Q., Hong, T., Yan, J., & Zhao, W. (2020). Integrating recurrent neural networks and reinforcement learning for dynamic service composition. *Future Generation Computer Systems*, *107*, 551–563. doi:10.1016/j.future.2020.02.030

Wang, H., Wang, J., Wang, J., Zhao, M., Zhang, W., Zhang, F., & Guo, M. (2018). Graphgan: Graph representation learning with generative adversarial nets. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1), 2508–2515. doi:10.1609/aaai.v32i1.11872

Wang, H., Yang, Z., & Yu, Q. (2017). Online reliability prediction via long short term memory for service-oriented systems. *2017 IEEE International Conference on Web Services (ICWS)*, 81-88. doi:10.1109/ICWS.2017.19

Wang, X., Ji, H., Shi, C., Wang, B., Ye, Y., Cui, P., & Yu, P. S. (2019). Heterogeneous graph attention network. *The World Wide Web Conference*, 2022-2032.

Wang, Y., Sun, H., Zhao, Y., Zhou, W., & Zhu, S. (2019). A heterogeneous graph embedding framework for location-based social network analysis in smart cities. *IEEE Transactions on Industrial Informatics*, *16*(4), 2747–2755. doi:10.1109/TII.2019.2953973

Wang, K., Gou, C., Duan, Y., Lin, Y., Zheng, X., & Wang, F. Y. (2017). Generative adversarial networks: introduction and outlook. *IEEE/CAA Journal of Automatica Sinica, 4*(4), 588-598.

Xia, B., Fan, Y., Tan, W., Huang, K., Zhang, J., & Wu, C. (2014). Category-aware API clustering and distributed recommendation for automatic mashup creation. *IEEE Transactions on Services Computing*, *8*(5), 674–687. doi:10.1109/TSC.2014.2379251

Ye, H., Cao, B., Peng, Z., Chen, T., Wen, Y., & Liu, J. (2019). Web services classification based on wide & Bi-LSTM model. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 43697–43706. doi:10.1109/ACCESS.2019.2907546

Yu, L., Zhang, W., Wang, J., & Yu, Y. (2017). Seqgan: Sequence generative adversarial nets with policy gradient. *Proceedings of the AAAI Conference on Artificial Intelligence*, *31*(1). Advance online publication. doi:10.1609/aaai.v31i1.10804

Zhao, K., Bai, T., Wu, B., Wang, B., Zhang, Y., Yang, Y., & Nie, J. Y. (2020). Deep adversarial completion for sparse heterogeneous information network embedding. *Proceedings of The Web Conference 2020*, 508-518. doi:10.1145/3366423.3380134

*Xiang Xie is a doctoral student in the School of Computer Science and Engineering and Hunan Key Laboratory for Service Computing and Novel Software Technology, Hunan University of Science and Technology, Xiangtan, China. Her research interests include heterogeneous information network, graph neural networks, and representation learning.*

*Jianxun Liu received his M.S. and Ph.D. degree in computer science from the Central South University of Technology in 1997 and Shanghai Jiao Tong University in 2003, respectively. He was a professor and dean of School of Computer Science and Engineering, Hunan University of Science and Technology. His current interests include Big Data, Business Intelligence, Service Computing and Cloud Computing. He has published more than 180 papers in peer reviewed journals, such as IEEE TPDS, IEEE TKDE, IEEE TSC, ACM T Web, etc. He served as a program committee member or program chair of the major conferences in his area (IJCAI, IEEE ICWS, IEEE SCC, APBPM, Trust Com, etc.).*

*Buqing Cao is currently a professor and vice dean in School of Computer Science and Engineering, Hunan University of Science and Technology, China. His research interests include service-oriented software engineering, service computing and social network. He has published more than 100 papers in the international conferences and journals, such as IEEE TSC, IST, JPDC, JWSR, ICWS, SCC, IEEE Big Data, APSCC, GCC, and HPCC. He is on the editorial boards of several journals and has served as a program committee member for multiple international conferences. He received his Ph.D. degree in computer science from Wuhan University, China, in 2011. He worked as a Postdoc. researcher from March 2015 to March 2016 in the Department of Computer Science and Computer Engineering at University of Arkansas, USA.*

*Mi Peng is an undergraduate student in the School of Computer Science and Engineering and Hunan Key Laboratory for Service Computing and Novel Software Technology, Hunan University of Science and Technology, Xiangtan, China. Her research interests include service classification, service recommendation, and representation learning. She has published 3 papers in CollabrateCom2020, CollabrateCom2021, and ICWS2021.*

*Guosheng Kang is a lecturer in the School of Computer Science and Engineering at Hunan University of Science and Technology, Xiangtan, China. His research interests include service computing and cloud computing, data-centric business process management, and data mining. He has published more than 30 papers. His publications have appeared in some well-known conferences and journals, such as IEEE ICWS, IEEE SCC, ICSOC, IEEE TSC, IEEE TNSM, FGCS, CCPE, etc. He received his Ph.D. degree in computer science from Fudan University, Shanghai, China.*

*Yiping Wen, Ph.D., is currently a professor in the School of Computer Science and Engineering and Deputy Director of the Department of Data Science at Hunan University of Science and Technology. His main research interests are cloud computing as well as big data, intelligent education, data mining and intelligent optimization, business process management and workflow technology. He is in charge of one national research project, and two provincial research projects, and has participated in five national research projects.*

*Kenneth K. Fletcher is an Assistant Professor of Computer Science at the University of Massachusetts Boston (UMass Boston) and the director of the Services Computing Lab. He received his Ph.D. in 2015 from Missouri University of Science and Technology in Computer Science with an emphasis in Software Design and Development. His research spans Machine and Deep Learning, Services and Cloud Computing, Software Engineering, and Additive Manufacturing. He has a patent and several publications in top conferences and journals in his research.*