



Article

Hidden-SAGE: For the Inference of Complex Autonomous System Business Relationships Involving Hidden Links

Haoyang Gao [†], Ning Li ^{*} and Yuancheng Xie [†]

Department of Computer Science, Harbin Institute of Technology, Harbin 150001, China; 22s130460@stu.hit.edu.cn (H.G.); 20b903060@stu.hit.edu.cn (Y.X.)

^{*} Correspondence: li.ning@hit.edu.cn

[†] These authors contributed equally to this work.

Abstract: Routing security is a crucial aspect of internet security. The main issues involved in routing security include Border Gateway Protocol (BGP) route leak and prefix hijacking. Currently, numerous solutions have been proposed for these issues, and significant breakthroughs have been achieved. However, these methods focus on visible data on the internet, overlooking the limited coverage of vantage points (VPs). Existing research indicates that attackers can cleverly design route announcements to evade detection by route collectors, thus executing routing attacks. Furthermore, many current methods for detecting route leaks rely on traditional business relationships between Autonomous Systems (AS), but the modeling of traditional AS business relationships is increasingly challenging to comprehensively cover business interactions between ASs. Therefore, we have developed Hidden-SAGE. A framework that extends AS-level internet topology and extracts complex business relationships between ASs from limited routing information. Hidden-SAGE utilizes graph neural networks to discover hidden AS links and employs random forests to infer complex business relationships between links. It successfully reduces visual bias caused by uneven VP distribution and constructs a more comprehensive AS-level internet rich-text topology. Compared to advanced inference algorithms, Hidden-SAGE performs better across various metrics and imposes fewer restrictions on the inference target.

Keywords: AS link; hidden AS link; BGP; AS relationship; complex AS relationship; GNN; GraphSAGE; random forest



Citation: Gao, H.; Li, N.; Xie, Y. Hidden-SAGE: For the Inference of Complex Autonomous System Business Relationships Involving Hidden Links. *Electronics* **2024**, *13*, 1617. <https://doi.org/10.3390/electronics13091617>

Academic Editor: Aryya Gangopadhyay

Received: 23 March 2024

Revised: 18 April 2024

Accepted: 19 April 2024

Published: 23 April 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The Internet comprises tens of thousands of Autonomous Systems (AS). As of 2023, the total number of AS in the entire Internet exceeded 70,000. An Autonomous System (AS) is a network unit internally comprising multiple routers that share a common intra-domain routing policy. AS can autonomously determine the intra-domain routing policy. Each AS is affiliated with an organization, which may be an Internet service provider, university, or company. The exchange of information among ASs is facilitated by the Border Gateway Protocol (BGP). Analysis and modeling of Internet AS topology play crucial roles in predicting and debugging routing paths. Estimates of AS-level topology have been employed in various research endeavors, encompassing the analysis of Internet topology attributes, inference of Internet hierarchy, construction of network topology generators for simulation, and assessment of the effectiveness and enhancements of new protocols [1]. There are many organizations that collect and make raw BGP data publicly available [2–4]. Presently, the construction of Internet AS topology primarily depends on the ASpath attribute in BGP raw data, thereby overlooking numerous links not observable in BGP raw data. Several studies [5–7] indicate that around 40% of AS links on the Internet are disregarded. These hidden AS links not only introduce incompleteness to the Internet’s AS-level topology but also pose significant risks to routing security. Milolidakis [8] discovered through

research that attackers can exploit these hidden links by intricately designing route announcements to evade the observation of route collectors, thereby executing attacks such as path spoofing [9] and prefix hijacking [10]. Additionally, the business relationship between the two ends of these hidden links remains unclear since the links cannot be observed in BGP routing data. AS business relationships are a critical factor in safeguarding against routing attacks such as route leak [11]. In the absence of certainty regarding AS business relationships, even if hidden AS links are discovered, it is challenging to ascertain whether a route containing hidden links poses a threat to routing security. Therefore, to mitigate carefully crafted routing attacks that exploit hidden links, it is imperative to unveil hidden AS links and infer the business relationships between them. This research is pivotal for enhancing the overall routing security of the Internet.

In conclusion, this paper aims to address two issues: (1) Inferring hidden AS links that are not observable by route collectors to expand the AS topology. (2) Inferring complex business relationships between hidden links. Regarding the inference of hidden links, only a limited number of researchers have attempted to develop algorithms. In 2014, Gregori et al. [12] proposed the MSC algorithm; however, significant challenges exist in applying this method to real network environments and achieving widespread adoption. This is attributed to the excessively high deployment cost of this method. In 2020, Jin [13] developed Toposcope, utilizing a decision-tree-based approach to infer hidden AS links. However, there is still room for further improvement in terms of its accuracy and recall rates.

Due to AS typically being unwilling to disclose their private business data, such as the internal network topology within an AS or a list of customers purchasing transit services on their network [14], obtaining accurate AS business relationships through publicly available datasets is impossible. Consequently, researchers have been prompted to model business relationships and develop algorithms for inferring them. To simplify heuristic reasoning development, Gao initially abstracted business relationships into three types [14]: p2c (provider-to-customer), p2p (peer-to-peer), and s2s (sibling-to-sibling). These three types of AS business relationships are commonly referred to as traditional AS business relationships. Over the past 20 years, numerous inference algorithms have been proposed for modeling traditional AS business relationships. Some advanced algorithms, such as ASRank [15], problink [16], Toposcope [13], AS-GCN [17], have demonstrated high accuracy and recall rates. However, in 2014, Giotsas [18] pointed out that modeling traditional business relationships overlooks more complex relationships, potentially introducing numerous issues in the study of inter-domain routing. Additionally, he presented a heuristic algorithm for inferring complex business relationships. Concurrently, Kastanakis pointed out in his research that relying solely on traditional business relationships is no longer sufficient for accurately inferring inter-domain routing policies. However, Giotsas's inference algorithm for complex business relationships cannot infer complex business relationships between hidden AS links.

We have decided to incorporate global Internet topology features to overcome the limitations of features. Thus, we integrate the graph neural network method into inferring hidden AS links. Graph neural network methods excel at processing complex graph-structured data and have shown outstanding performance in tasks such as node classification, link prediction, and image clustering [19]. Among various graph neural network models, we opt for the GraphSAGE [20] model. This model efficiently learns to generate node embeddings for previously unseen data by sampling and aggregating features from the local neighborhoods of nodes. It exhibits excellent performance in handling large-scale tasks such as image processing. In this article, building upon Giotsas's classification model for complex AS business relationships, we concentrate on full transit, partial transit, and peering links. We introduce novel features and develop a new graph-sampling network framework. This framework comprehensively captures relevant feature information from the training set to the test set. Utilizing the inference framework proposed in this article,

we can infer more complex business relationship links. The primary contributions of this study are summarized as follows.

- (1) We address hidden AS links by constructing an experimental dataset that ensures consistency between existence and observability. Simultaneously, we design a series of features, and these datasets and features are integrated into the graph sampling network framework. Inference is conducted on hidden AS links in datasets from 2014 and 2023. To our knowledge, this study is the first to utilize GraphSAGE to address the challenge of inferring hidden AS links.
- (2) In addressing complex business relationship issues, we introduce a series of empirically effective and self-proposed essential features. We utilize the random forest algorithm to infer p2c, p2p, and partial transit business relationships.
- (3) We developed a framework called Hidden-SAGE. This framework aims to infer hidden AS links and the intricate business relationships between them. We named this framework Hidden-SAGE because it focuses on “hidden” AS links that are overlooked by the route collector, and it utilizes GraphSAGE. We conducted comprehensive experiments to compare the methods proposed in this article with existing state-of-the-art methods.

The remainder of this paper is structured as follows: Section 2 presents the current state of research in the areas of hidden AS link inference and AS business relationship inference. Section 3 presents the details of our proposed Hidden-SAGE framework. Section 4 outlines the experiments conducted, along with the analysis and interpretation of the results. Finally, in Section 5, we summarize the entire paper and discuss future directions.

2. Related Work

In the realm of inferring hidden AS links, in 2014, Gregori et al. [12] proposed the MSC algorithm to find the optimal and minimal VP positions to observe all AS links. However, the actual deployment cost was prohibitively high. In 2020, Jin developed the Toposcope [13] framework, utilizing a decision tree to infer hidden AS links, and identified 30,000 possible hidden links. Nevertheless, its accuracy and recall rates were only 83% and 72%, respectively, indicating room for improvement.

In the domain of AS business relationship inference, in 2001, Gao Lixin [14] pioneered the modeling of AS business relationships. The model classifies AS business relationships into three types: (1) Provider-Customer (p2c), (2) Peer-to-Peer (p2p), and (3) Sibling-to-Sibling (s2s). Additionally, Gao proposed the hypothesis that BGP routing paths should follow the valley-free path principle. This implies that an AS path is composed of zero or more c2p or s2s links, followed by zero or one p2p link, and then followed by zero or more p2c or s2s links. The shape of this path consists of an uphill path, a downhill path, or a combination of both. We refer to Gao’s modeling of AS business as traditional AS business relationships. This modeling played a crucial role in subsequent algorithms for business relationship inference.

In 2013, Luckie [15], without relying on the valley-free path principle, proposed new hypotheses for the inter-domain structure of the Internet: (1) AS establishes provider relationships to achieve global reachability; (2) there exist peer groups of AS at the top of the hierarchy; and (3) there are no loops in p2c links. Based on these hypotheses, Luckie proposed a new inference algorithm for traditional AS business relationships called ASrank, achieving high accuracy. Additionally, Luckie introduced the concept of the customer cone to assess the significance of a specific AS in the Internet.

In 2019, Yuchen Jin [16] and other researchers addressed the challenging issue of inferring business relationships in hard links, including non-valley routing, limited visibility, and unconventional peering practices. They proposed the ProbLink inference algorithm. ProbLink revealed critical AS interconnection features originating from random information signals. Compared to AS-Rank, ProbLink reduced the error rate for all links by 1.6 times. For various types of hard links, the error rate was reduced by 6.1 times.

In 2020, Zitong Jin [13] and other researchers developed the Toposcope algorithm. This method addresses the issues of limited coverage and biased concentration of vantage points (VPs). It utilizes ensemble learning and Bayesian networks to alleviate observation bias. Compared to ASRank, Toposcope reduces inference errors by 2.7–4 times. It discovered relationships for approximately 30,000 upper-level hidden AS links. Moreover, it remains more accurate and stable even under more incomplete or biased observations.

In 2022, for the inference of S2S and X2X relationships, PENG [17] introduced graph neural network technology into AS business relationship inference, achieving an accuracy rate of over 90%.

Currently, methods for inferring traditional business relationships have achieved considerable success. However, with the development of the Internet, the traditional classification of business relationships cannot fully describe the routing policies between ASs. Giotsas proposed in 2014 [18] that oversimplified business relationship modeling introduces many issues into inter-domain routing policy research, a viewpoint confirmed by the work of many other Internet researchers [21–27]. Therefore, Giotsas proposed a classification model for complex business relationships in 2014, including four types of business relationships: (Full) transit [28]: an AS provides access to its peers, customers, and providers to another AS. Peering [28]: two ASs mutually provide access to their customer routes. Partial transit [29]: an AS provides access to its peers and customer routes to another AS but does not provide access to providers. Hybrid: an AS provides a combination of full transit, partial transit, or peering based on interconnection locations to another AS. Additionally, a heuristic algorithm was proposed for inferring complex business relationships, with positive predictive values of 92.9% and 97.0% for partial transit and hybrid relationships, respectively. Although this method has high accuracy, it still has certain limitations. The method relies on data flow and can only infer complex business relationships of visible links in BGP raw data.

In 2023, Kastanakis attempted to replicate the work of Wang and Gao from 2003 to discover the impact of Internet topology changes over the past 20 years on routing policies. The study also aimed to assess the validity of the research results of Wang and Gao in today's Internet [30]. Kastanakis made two significant findings in the research. Firstly, Kastanakis observed that, during route import, ASs no longer exhibit strong consistency in setting locpref values [31] based on their business relationships. Locpref is a non-transitive numerical BGP attribute that denotes the preference of a certain route. Additionally, there is a notable occurrence of setting multiple locpref values for the same neighbor. Kastanakis provided two explanations for this phenomenon: the absence or lack of popularity of certain business agreements in the early stages and the existence of complex business relationships between ASs. Secondly, Kastanakis discovered a considerable portion of selective announcement prefixes in the current Internet. These selective announcement prefixes persist for an extended period and are prevalent in both exports to provider ASs and exports to peer ASs. These findings indicate that relying solely on traditional business relationships is insufficient for accurately inferring inter-domain routing policies.

3. Hidden-SAGE

In this section, we will provide detailed descriptions of our two tasks: inference of hidden AS links and inference of complex business relationships. In the hidden AS links inference part, we devised an inference framework based on graph neural networks, utilizing existing features to infer hidden links. Specifically, we employed a graph sampling network model as the graph neural network layer, and we will explain the rationale behind this choice shortly. Additionally, we incorporated nine node features as inputs to the graph neural network model, some of which were previously used by other researchers, and some were designed by us. The reasons for selecting these features will be elucidated in the following. The workflow of the entire framework involves (1) extracting node features, (2) training the model with both graph structure and feature data, and (3) minimizing loss to obtain accurate link classification results. In the part concerning the inference of complex

business relationships, we utilized the random forest algorithm, leveraging node features designed by other researchers and ourselves to infer these relationships.

3.1. Inference of Hidden AS Links

3.1.1. Training Dataset

According to Jin's definition, formally, if a pair of consecutive links $\langle \ell_{AC}, \ell_{CB} \rangle$ can be observed in any path, we refer to $\langle \ell_{AC}, \ell_{CB} \rangle$ as a detour pair. This represents a two-hop detour path from A to B. On the other hand, if a link ℓ_{AB} exists but can only be observed by a single vantage point (VP), we refer to it as a singleton link. A and B are also referred to as endpoints of $\langle \ell_{AC}, \ell_{CB} \rangle$, or ℓ_{AB} .

In the process of constructing positive and negative samples, randomly selecting a detour pair $\langle \ell_{AC}, \ell_{CB} \rangle$ and checking if ℓ_{AB} can be observed (only one VP) is unreliable. This is because when ℓ_{AB} is not discovered, we cannot conclusively determine whether it is due to the non-existence of ℓ_{AB} on the Internet or its unobservability. Here, we refer to Jin's assumption: VPs will report all their neighboring ASs. Therefore, when constructing training samples, we restrict the scope to detour pairs that start or end with a full VP to ensure that the existence and observability of the link are consistent.

For training data used to infer hidden links, we construct two sets of training data. In the first set, we follow the Toposcope approach and find all detour triplets with a full VP as endpoints. For each such detour triplet, $\langle A, C, B \rangle$, if the link AB is observed as a singleton link and $\langle A, C, B \rangle$ is observed by different VPs, we add $(\langle A, B \rangle, 1)$ as a positive sample to our training set. If AB is not observed as a singleton link, we use $(\langle A, B \rangle, 0)$ as a negative training sample. This allows us to compare with the Toposcope method. In the second set, we directly consider singleton links as positive samples and links with full VPs as endpoints but not present in the AS path as negative samples. This allows us to evaluate our method after removing the constraint of one-hop detour paths with full VP endpoints. After determining positive and negative samples, we remove the positive samples from the graph data to prevent information leakage.

3.1.2. Feature Selection

In the subsequent feature selection process, for node attribute features, we selected some features that have been chosen and proven effective by network researchers [13], and we also proposed some features ourselves. In addition to this, we leveraged the self-learning characteristics of graph neural networks to obtain high-dimensional graph structure features, achieving the best inference results by combining them with node attribute features.

For the node features of hidden links, we removed features related to triplets based on the features used to infer hidden links in Toposcope [13], as our training set is based on links and is unrelated to triplets. Toposcope has utilized features including AS Degree, AS Transit Degree, AS Hierarchy, AS Pagerank, and Number of High-Hierarchy Neighbors, demonstrating their effectiveness in inferring hidden links. We will employ these features. In addition to these features, this paper introduces attributes such as AS type, AS geographical location, and AS customer cone size into the node attribute features. The introduced features are shown in Table 1. Ablation experiments were designed to demonstrate the validity of these features, elaborated upon in detail in Section 4.5.

Table 1. Features for inferring hidden links.

Features	Description
AS Degree	Number of Neighbors for AS
AS Transit Degree	Extract triplets centered around AS i and determine the count of AS i's unique neighbors in the transmission path.

Table 1. Cont.

Features	Description
AS Hierarchy	Hierarchy of AS in the Internet
AS Pagerank	A link analysis algorithm that assigns numerical weights to each element in a hyperlinked document set with the goal of “measuring” its relative importance within that collection.
AS Organizational Type	The organizational types of AS, with three common types: Transit/Access AS, Content AS, and Enterprise AS.
AS Customer Cone Size	Metrics proposed by CAIDA to measure the importance of AS
AS Geographic Location	Country of AS registry
Number of High-Hierarchy Neighbors	Number of neighboring AS in the core and high layers

3.1.3. Model Framework

In this section, we introduce our design of the graph neural network model. The reasons for using a graph neural network for hidden link inference are as follows: (1) Graph neural networks excel in handling link prediction tasks; (2) We aim to leverage the graph neural network to generate higher-dimensional features to assist in our predictions.

The model we use is illustrated in Figure 1. In contrast to traditional methods for inferring hidden AS links (e.g., Toposcope), we introduce a graph neural network model to address this problem. To our knowledge, our approach is the first to employ a graph neural network for hidden AS link inference. In the subsequent task processing, we draw inspiration from the work of Peng, wherein specific node vectors are concatenated to represent corresponding links following the acquisition of node vector representations. Subsequently, the obtained vector representation of the link undergoes processing through a fully connected layer and utilizes the sigmoid function to obtain the classification result of that link. Both the graph neural network model and the fully connected model utilized in our framework are founded on prior research. Our model mainly consists of the following components:

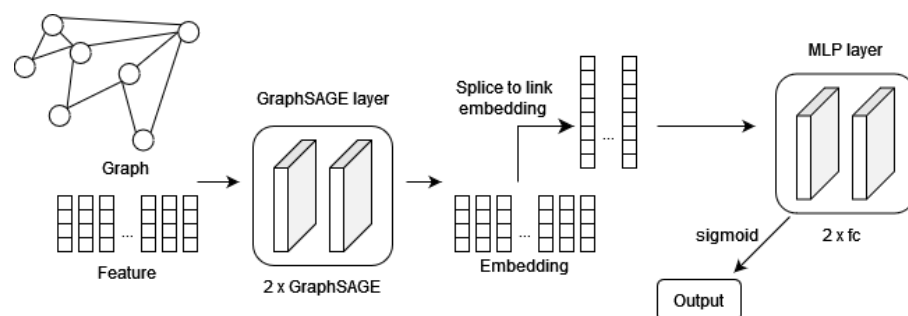


Figure 1. Hidden link inference model framework.

- **Input Layer**

In the input layer, we input the raw data and preprocess them to generate graph data and data for downstream feature statistics. The graph data are represented as an undirected graph $G = (V, E)$, where V is the set of ASs, and $E \in V \times V$ is the set of edges. v_i represents an AS, and $e_{ij} = (v_i, v_j)$ is an edge with AS v_i and v_j as endpoints. In graph neural networks, graph data are input in the form of an adjacency matrix. The adjacency matrix A is an $n \times n$ matrix, where $A_{ij} = 1$ if $e_{ij} \in E$, and $A_{ij} = 0$ if $e_{ij} \notin E$. For the hidden link prediction task, we use all detour paths as the graph data for information propagation. In this problem, $E \in V \times V$ represents the remaining edges after removing all positive samples. This ensures that the edges

used for training and testing do not appear in the graph used for message passing. Otherwise, there would be a situation where the predicted edges are already known to exist, rendering the prediction meaningless. This also simulates the observation bias in the Internet due to the lack of VPs.

- **Feature Layer**

In the feature layer, we employ the data from the input layer to statistically analyze and summarize the original features. We construct node feature vectors, which consist of the crucial features mentioned in Section 3.1.2. $R^{n \times d}$ represents the feature matrix, where each node in the graph has attributes $X \in R^{n \times d}$. $x_v \in R^d$ represents the feature vector of node v .

- **GraphSAGE Layer**

In the previous sections, we introduced the idea of using the GraphSage [20] model. GraphSAGE is a general inductive framework that efficiently generates node embeddings for previously unseen data using node feature information, such as text attributes. GraphSAGE does not train a separate embedding for each node but learns a function to generate embeddings by sampling and aggregating features from a node's local neighborhood. Additionally, we referred to the work of Jin [32], which indicated that stacking two layers of graph neural network models results in relatively better overall performance. Hence, we stacked 2 layers of GraphSAGE models. The pseudocode for generating node embeddings using GraphSAGE is presented in Algorithm 1. First, each node $v \in V$ aggregates its representations of neighboring nodes $\{h_u^{k-1}, \forall u \in N(v)\}$ into a vector. After aggregating neighbor feature vectors, GraphSAGE aggregates the current h_v^{k-1} node and neighbor vector $h_{N(v)}^{k-1}$. Subsequently, this concatenated vector is transformed into a vector for downstream tasks (i.e., $h_v^k, \forall v \in V$) through a fully connected layer with a non-linear activation function σ .

Algorithm 1 GraphSAGE embedding generation algorithm.

Input: Graph $G(V, E)$; input features $x_v, \forall v \in V$; depth K ; weight matrices $W^k, \forall k \in \{1, \dots, K\}$; non-linearity σ ; differentiable aggregator functions $AGGREGATE_k, \forall k \in \{1, \dots, K\}$; neighborhood function $N : v \rightarrow 2^V$

Output: Vector representations z_v for all $v \in V$

```

1:  $h_v^0 \leftarrow x_v, \forall v \in V$ 
2: for  $k = \{1 \dots K\}$  do
3:   for  $v \in V$  do
4:      $h_{N(v)}^k \leftarrow AGGREGATE_k(\{h_u^{k-1}, \forall u \in N(v)\})$ 
5:      $h_v^k \leftarrow \Sigma(W^k \times CONCAT(h_v^{k-1}, h_{N(v)}^k))$ 
6:   end for
7:    $h_v^k \leftarrow h_v^k / \|h_v^k\|_2, \forall v \in V$ 
8: end for
9:  $z_v \leftarrow h_v^k, \forall v \in V$ 

```

- **MLP Layer**

The MLP layer serves as the “classifier” in the entire model. We stack two layers of fully connected layers, where the first layer transforms node embeddings into edge tensors, and the second layer is responsible for classifying edges based on the edge tensors. When inferring hidden link issues, we employ the sigmoid function because it exhibits good performance in handling binary classification. Let Z represent the output of the GraphSAGE layer. For binary classification problems, we use the following expression:

$$Y_{ij} = \text{sigmoid}(\text{Linner}(Z_i, Z_j)) \quad (1)$$

In this context, Z_i and Z_j are the respective vectors output by the GraphSAGE layer for the two end nodes of the edge e_{ij} . Y_{ij} is a probability vector used for the classification

of the corresponding edge labels.

For semi-supervised binary classification, we employ binary cross-entropy loss as the objective function, represented as

$$\mathcal{L}_{BCE}(Y, Y') = -\frac{1}{n} \sum_{i=1}^n y_i \log(y'_i) + (1 - y_i) \log(1 - y'_i) \quad (2)$$

Here, $\mathcal{L}_{BCE}(Y, Y')$ represents the binary cross-entropy loss across the entire dataset, where n is the number of samples, y_i is the actual label of the i -th sample, and y'_i is the probability predicted by the model for the i -th sample.

- **Output**

In reporting the results, we uniformly set the epoch to 200, conducting 10 experiments and averaging the results to compile various metrics.

3.2. Inference of Complex Business Relationships

3.2.1. Basic Ideas

In the task of inferring complex business relationships, we focus on the classification of complex business relationships based on Giotsas [18], specifically emphasizing the categories of p2c, p2p, and partial transit. We do not focus on hybrid relationships, and our reasons for this are as follows: Firstly, inferring hybrid relationships requires the introduction of more granular data. Since hybrid relationships constitute a small portion of the Internet, this would make the model more complex and potentially affect the inference accuracy. Secondly, for hidden links, it is highly probable that obtaining granular data flows is not feasible. Therefore, inference can only be performed at the Autonomous System (AS) level. Thirdly, Jin pointed out that in hidden links, the majority of business relationships are p2p, with a smaller portion being p2c. However, based on Giotsas' definition, we believe that a portion of hidden links is likely to be partial transit. Therefore, we specifically focus on p2c, p2p, and partial transit. Therefore, our emphasis is on the categories of p2c, p2p, and partial transit.

3.2.2. Features of Complex Business Relationships

For link features associated with complex business relationships, to demonstrate the effectiveness of these features in solving the inference problem related to complex business relationships, we adopt the effectiveness exploration method proposed by Peng [17]. This involves analyzing the effectiveness of features by calculating the difference in feature values between the source Autonomous System (AS) and the target AS. The expression is given below.

$$\Delta = |f_{AS1} - f_{AS2}| \quad (3)$$

In this context, f_{AS1} and f_{AS2} represent the feature values of the left and right endpoints, respectively, and Δ is the difference in endpoint feature values. A departure from Peng's feature processing approach, our feature selection directly serves as the features of the link, while Peng's features are designed for nodes and are subsequently aggregated into link features in downstream tasks.

When compiling features for complex business relationships, we utilized the list of complex business relationships published by Giotsas [18] in 2014. The BGP raw data employed were the data used to infer this list of relationships.

- **Degree and Transit Degree**

Degree is one of the fundamental metrics in graph theory. In the undirected topology of the Internet, the degree of a node refers to the number of neighboring nodes directly connected to that node. A higher degree value indicates greater importance of the node in the graph. Transit degree, introduced by Luckie [15] in 2013, is an indicator reflecting the service scale of an Autonomous System (AS) and the degree of collaboration with other ASs. By extracting triplets with AS i as the center, the transit

degree of AS i is obtained by counting the unique neighbors of AS i in the transit paths. For example, given triplets (AS1, AS2, AS3), (AS1, AS2, AS4), (AS5, AS2, AS6), where AS2 is in the middle position, and its unique neighbors are AS1, AS3, AS4, AS5, and AS6, the transit degree of AS2 is 5. ASs such as AS1, AS3, AS4, AS5, and AS6, with a transit degree of 0 as they are not observed in the middle positions of triplets, are referred to as stub ASs located at the outermost layer of the Internet. Both degree and transit degree effectively reflect the importance of a specific AS on the Internet and are crucial graph features for subsequent algorithm design.

- Distance to Clique AS

Clique AS refers to a cluster of Autonomous Systems (ASs) situated at the top level of the Internet hierarchy. Within the Clique AS cluster, ASs mutually exchange traffic free of charge, without any associated fees (akin to a P2P relationship). Clique AS is often regarded as the center of the Internet, as researchers widely agree on an assumption: ASs at the topmost level of the Internet hierarchy have numerous customers among them, making these high-level ASs more likely to engage in peer-to-peer relationships, fostering mutual benefits. Conversely, for lower-level ASs, forming customer-to-provider (C2P) relationships with higher-level ASs to achieve global accessibility is considered a more optimal access strategy. Acquiring the distance feature from a specific AS to Clique AS is essentially aimed at measuring the distance of that AS to the Internet's center.

Initially, we obtain the list of Clique AS using the method proposed by Luckie [15]. Next, we extract the AS path attributes from the BGP raw data to construct an undirected graph. Subsequently, we calculate the shortest path distance from each AS in the undirected graph to every member AS in Clique AS, taking the average as the distance feature from that AS to Clique AS. We employ Equation (1) to assess the effectiveness of this feature, using the 2014 validation dataset to evaluate the three types of AS links. The distribution of distances to the Clique AS, as shown in Figure 2, significantly highlights the differences in the types of relationships: full transit, peering, and partial transit. The distinction between peering and p2c, and partial transit is quite evident. While the distributions of p2c and partial transit are closer, they can still be clearly differentiated. This occurs because, according to Giotsas's definition [18], partial transit is considered a special case of p2c, leading to a tendency for similarity in feature distribution, yet with discernible differences.

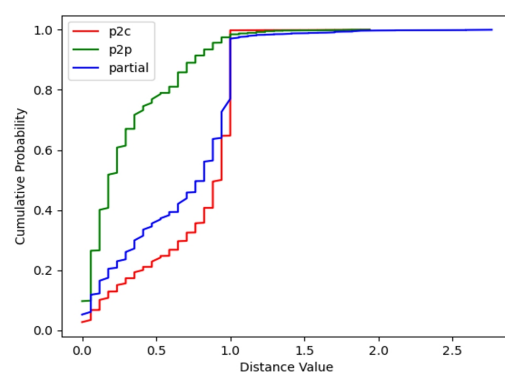


Figure 2. Probability distribution of distances to Clique AS.

- AS Organizational Types

The organizational type of an AS determines its function and influences its relationships with neighbors. Common AS organizational types include three categories: Transit/Access AS, Content AS, and Enterprise AS. Transit/Access AS serves as a provider of transit and/or access services. Content AS provides content hosting and distribution services. Enterprise AS encompasses various organizations, universities, and companies. Typically, Transit/Access AS provides transit/access services

to Content AS and Enterprise AS, reflecting well-established business relationships among ASs. We obtained the AS organizational type dataset publicly available from CAIDA [33] and conducted a statistical analysis of the AS organizational types included in the three categories of business relationships. The experimental results are presented in Figure 3, where Transit/Access AS is labeled as 0, Content AS as 1, and Enterprise AS as 2.

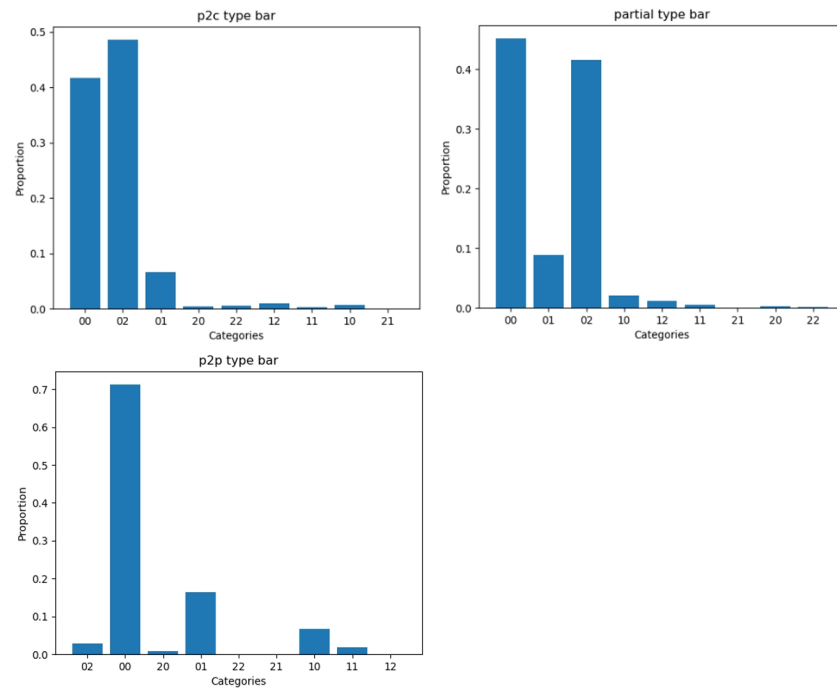


Figure 3. Bar chart of AS organizational type distribution.

- AS Hierarchy**
 Carmi [34] proposed in 2007 that the Internet follows a certain hierarchical structure. For different AS links, we focus on which hierarchy each AS at the end of the link belongs to and calculate the proportion based on business relationships. Yang [35] once proposed a model that divides the Internet into four levels: (1) Core Layer: All ASs in the Clique AS form the core layer. Because these ASs have the highest transit degree, we label them as 1. We use the Clique AS list inferred from CAIDA. (2) High Layer: ASs that are direct customers of the core layer and have a neighbor degree greater than 100. (3) Stub Layer: ASs that are non-direct customers of the core layer and have no customers, i.e., ASs with a transit degree of 0. (4) Low Layer: ASs other than the above four categories. As shown in Figure 4, we can observe that p2c and partial relationships predominantly occur between ASs in the 3rd and 4th layers, while p2p relationships primarily occur between ASs in the 2nd and 3rd layers.
- Number of High-Hierarchy Neighbors**
 ASs located at higher levels often occupy the core of the Internet. Here, we refer to ASs situated in the first and second layers as high-level ASs. For an AS, the number of high-level neighbors indirectly indicates the magnitude of traffic it transmits on the Internet. Hence, we calculate the difference in the number of high-level neighbors between the two end nodes in three types of business relationship links. The statistical results are shown in Figure 5, indicating that this feature can distinctly differentiate between p2c and p2p relationships.

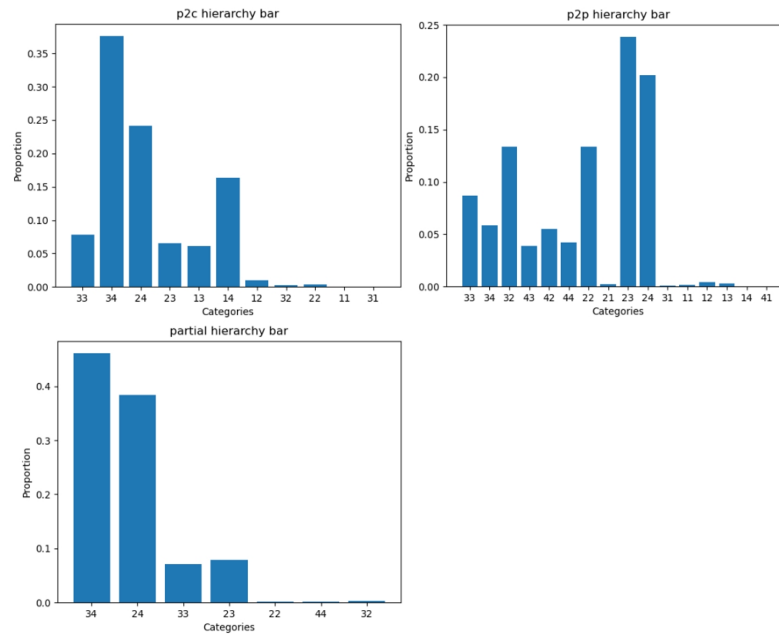


Figure 4. Bar chart of AS node hierarchy distribution.

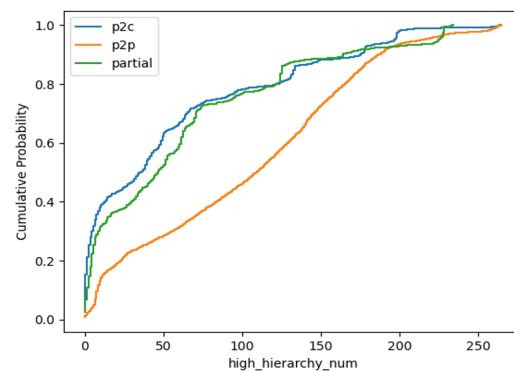


Figure 5. Probability distribution of numbers of high-hierarchy neighbors.

- **AS Pagerank**

Pagerank is a link analysis algorithm that assigns numerical weights to each element in a hyperlinked document set (such as the World Wide Web), with the aim of “measuring” its relative importance in that set. The algorithm can be applied to any collection of entities with mutual references and citations. The numerical weight assigned to any given element E is called the Pagerank of E , denoted as $PR(E)$ [36]. We calculated the Pagerank differences between nodes in three types of business relationships, and the statistical results are shown in Figure 6. From the graph, we can observe that the Pagerank feature can distinctly differentiate between partial transit links and p2c links.

- **AS Geographic Location**

ASs have geographical locations in the real world, as proposed by Li [37]. For convenience, ASs typically purchase access services from ASs in the same location when connecting to the Internet. Subsequently, the access service ASs purchase transit services from ASs in another location to achieve global Internet connectivity. Therefore, we believe that geographical location can reflect business relationships between ASs to some extent. This feature is also applied for the first time in relationship inference. We obtained the AS geographic location dataset publicly available from CAIDA [38]. We conducted a statistical analysis of the geographic locations of nodes in each AS link, and the experimental results are shown in the graph. We observed that, for p2c links, the proportion of nodes at both ends with the same geographic location is 77.9%, for p2p links, it is 30.8%, and for partial

transit links, it is 71.2%. From the data, we can see that p2c and p2p links exhibit significant differences in the geographic location feature.

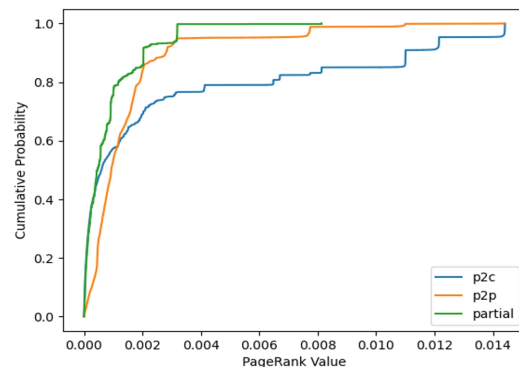


Figure 6. Probability distribution of AS pagerank values.

3.2.3. Training Samples and Inference Method

All our training samples are derived from the inference results of complex business relationships by Giotsas in 2014, published by CAIDA, and the traditional business relationship inference results in 2014. We first remove partial transit and Hybrid links from all p2c links to obtain purely transit p2c relationships. Subsequently, we select p2c, p2p, and partial transit links. We use all partial transit links and randomly select p2c and p2p links to ensure a 1:1:1 ratio in the sample quantities for the three relationship types. We split the dataset into a training set and a testing set in a 7:3 ratio. After constructing the training samples, we train a random forest model with the training set and then test the training results using the testing set.

4. Experiments

4.1. Datasets

These files underwent preprocessing by CAIDA, including discarding paths containing sets of ASs, compressing path padding (i.e., converting AS paths from “A B B C” to “A B C”), discarding poisoned paths (i.e., paths in which an ASN occurs more than once and is separated from it by at least one other ASN, or paths in which Clique ASs are separated), discarding paths containing unassigned ASs, and discarding paths containing ASNs belonging to IXPs. We scan these paths and extract all paths containing IPv4 prefixes, discarding those containing IPv6 prefixes based on the prefix attribute in ASpath. As of now, our access is limited to the inference results of AS complex business relationships from CAIDA in 2014. Simultaneously, for the purpose of comparing changes in hidden AS link characteristics between 2014, a decade ago, and 2023, the present year. We selected files from March 2014 and March 2023.

Validation Data

In the case of hidden AS link inference, we utilize AS links observed by only one VP as the ground truth dataset for validation. In the case of complex business relationship inference, we rely on the list of AS complex business relationships inferred using Giotsas’s method, as published by CAIDA, and validate it as a ground truth dataset.

4.2. Inference of Hidden AS Links

We applied our method to the datasets of 2014 and 2023, conducting experiments using two types of samples. We removed all observable positive samples’ AS paths. If an AS path from the remaining paths did not include the ASs in the sample, we added the AS path back to the original data and removed the link from the positive samples. The processed AS path data were then applied to both Toposcope and our method. The experimental results for hidden link inference are shown in Figure 7.

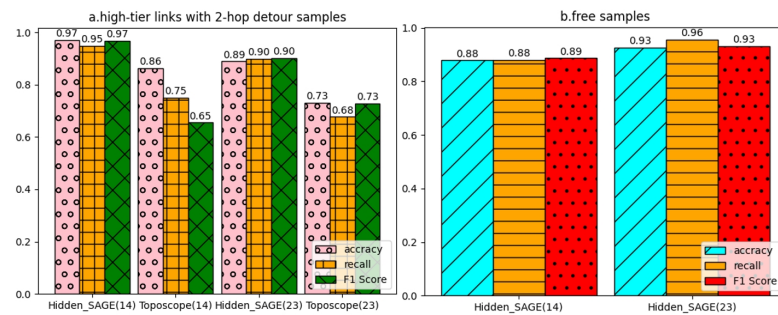


Figure 7. (a) Inferred results with more constraints. (b) Inferred results with less constraints.

From the experimental results, we can observe that our method outperforms Toposcope in various metrics. Additionally, even with a smaller sample size, our method maintains a high level of accuracy. For different sample restrictions, in 2014, our method performed better in samples with more restrictions, while in 2023, our method excelled in samples with fewer restrictions. According to Kastanakis [30], the Internet hierarchy tends to flatten over the years. Our experimental results indicate that the features selected in our method align better with samples that have more restrictions in the past. As the Internet evolves, the features proposed in our method become more suitable for inferring the existence of modern Internet links, with broader coverage.

4.3. Inference of Complex AS Business Relationships

To validate our ability to infer the business relationships of hidden links more accurately, we extracted links with p2p, p2c, and partial transit relationships in 2014 at a 1:1:1 ratio, totaling 8985 links. We iterate through all the AS paths, and if an AS path contains any of these 8985 links, we remove that AS path from our dataset. If an AS path does not contain any of these 8985 links, we keep the AS path. Subsequently, we used Toposcope and our method separately to infer the existence and business relationships of these links. The experimental results are shown in Figure 8.

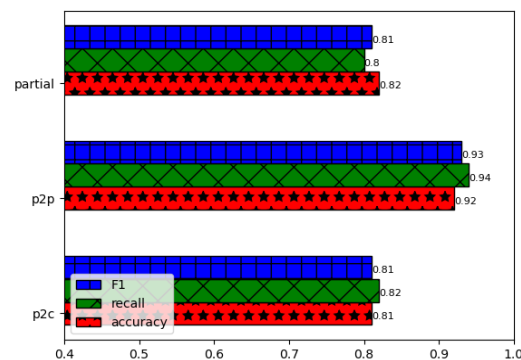


Figure 8. Results for inference of complex AS business relationships.

Our algorithm inferred 7736 links out of 8985 samples. When inferring complex business relationships, we removed the portion of these links that existed in our training set for complex business relationships. This prevents information leakage; then, we proceeded to infer complex business relationships for the remaining links. Our overall accuracy for inferring complex business relationships is 85%. For p2c inference, the accuracy, recall, and F1 Score are 81%, 82%, and 81% respectively. For p2p inference, the three metrics are 92%, 94%, and 93%. For partial transit inference, the three metrics are 82%, 80%, and 81%.

Toposcope only inferred 45 links among these, and 42 of these links had business relationships consistent with the training set. In addition to these links, Toposcope inferred 2594 hidden links between upper-layer Ass with a business relationship accuracy of 93.90%.

Jin believes the reasons for the insufficient number of inferences by Toposcope are as

follows: (1) Upper-layer links are more likely to be observed. (2) The impact of upper-layer links on Internet routing is higher than that of lower-layer links. (3) Due to data sparsity, finding a bypass pair corresponding to a singleton link in the lower layer is challenging. (4) Learning from (or inferring) more bypass couples would increase costs and might result in a more complex but less accurate classification model (or inference results). Therefore, Jin used the K-means algorithm to cluster some ASs with similar features, all belonging to the upper-layer ASs defined by Jin and exhibiting behavior similar to complete VP. Toposcope only infers hidden links and business relationships between these ASs. However, most ASs in the links in our samples are not within this range, leading to their omission by Toposcope.

Our experimental results indicate that Toposcope has limited coverage in inferring hidden links, while our method can infer hidden links between ASs without being restricted by AS hierarchy. Additionally, in terms of inferring business relationships, while Toposcope can only infer traditional business relationships for a limited number of hidden AS links, our method extensively infers complex business relationships among the most hidden AS links. This highlights our superiority over Toposcope in two aspects: (1) Our method can infer a greater number of business relationships between hidden AS links, demonstrating its generalizability. (2) Our method can infer various types of business relationships, specifically complex ones, among hidden AS links. This suggests that our method can precisely define the types of business relationships between hidden AS links. Improving the accuracy of the types of business relationships between ASs can be very helpful in solving route leaks.

4.4. Replacement Experiment

To demonstrate that our model selection achieves optimal results, we internally replaced the original model in our method framework with other state-of-the-art models and then recorded various prediction metrics.

4.4.1. Inference of Hidden AS Links

In the replacement experiment for hidden link inference, we used the 2023 dataset with unrestricted links. We describe this dataset in detail in Section 3.1.1. We performed metric statistics for three of the most popular graph neural network models: GCN, GraphSAGE, and GAT. We chose these three traditional graph neural network models as components in our framework for comparison because the more advanced graph neural network algorithms currently available tend to impose greater restrictions on the dataset and have more parameters. We aim for our approach to perform effectively on less restrictive datasets that closely resemble real network environments. Therefore, we only compare these three traditional and popular graph neural network models to observe which one achieves the best results when integrated into our framework. The experimental steps were similar to the ones mentioned earlier, with each model trained for 200 epochs, and the results were averaged over 10 experiments. The experimental results are shown in Figure 9. From the results, we can see that GraphSAGE outperforms the other two models in terms of accuracy and F1 score. This demonstrates the excellent performance of GraphSAGE in handling large graphs.

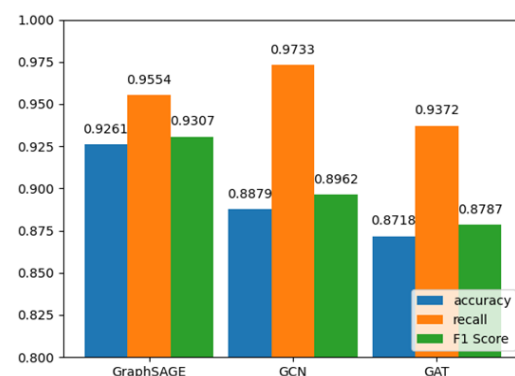


Figure 9. Replacement Experiment Results for hidden link inference.

4.4.2. Inference of complex AS business relationships

In the replacement experiment for inferring complex AS business relationships, we used the 2014 dataset. Five currently popular machine learning methods were selected, and the experimental results are shown in Figure 10. Our rationale for comparing the effects of these general machine learning methods here aligns with the reasons stated in Section 4.4.1. We aim to avoid imposing additional restrictions on the dataset and complicating the problem by increasing the number of parameters. Furthermore, in the domain of AS business relationship inference, researchers are still exploring how to optimize the utilization of machine learning methods. This challenge arises from the confidentiality and complexity of AS business relationships, which prevents comprehensive coverage of all relationships between ASs, thereby complicating feature design. Presently, only Problink and Toposcope utilize machine learning methods for inferring AS business relationships. Additionally, these two algorithms only infer conventional business relationships. In this paper, we pioneer the application of machine learning methods to infer complex business relationships, investigating the feasibility of such inference. The experimental data demonstrate that inferring complex AS business relationships using common machine learning methods can yield respectable results. Moving forward, we can concentrate on feature design to enhance the accuracy of the inference algorithm and other metrics. Based on the experimental results, it is evident that both random forest and XGBoost perform similarly and outperform other machine learning methods. However, XGBoost only marginally outperforms random forest in terms of accuracy and F1 score for inferring p2p types. In all other aspects, random forest outperforms XGBoost; therefore, we select random forest.

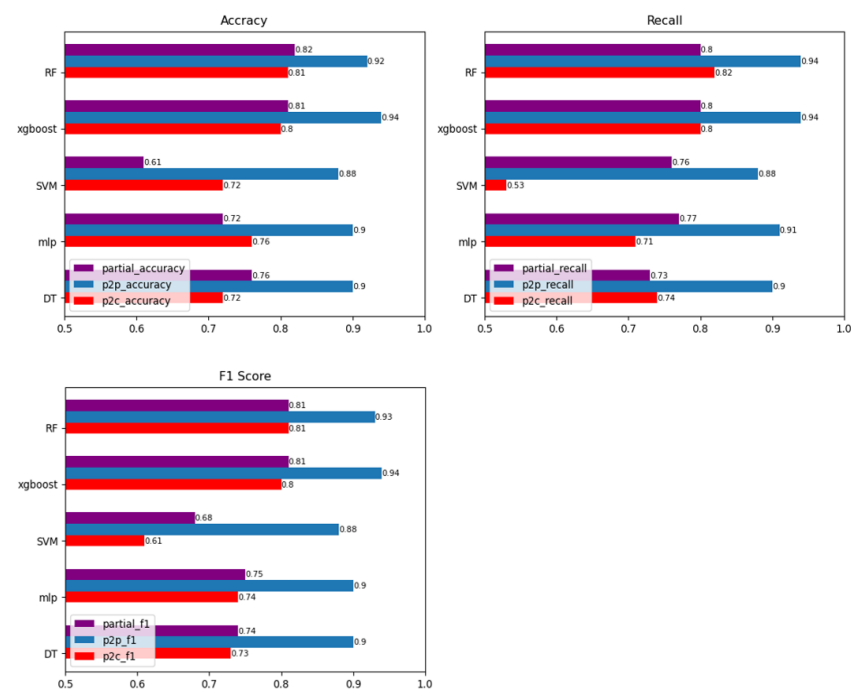


Figure 10. Replacement Experiment Results for inference of complex AS business relationships.

4.5. Ablation Experiment

To explore the importance of features in inferring the two problems, we conducted ablation experiments specifically on the features for both problems. For hidden AS link inference and complex AS business relationship inference, we employ the same procedure during ablation experiments. Initially, we maintain the model and dataset unchanged. Subsequently, we individually remove one feature and observe the resultant change in method accuracy. For hidden link inference, we used a dataset with fewer samples from 2023 for ablation experiments. For complex business relationship inference, we conducted

ablation experiments using the 2014 dataset which we mentioned in Section 4.3. Accuracy changes were recorded in the experiments.

4.5.1. Hidden Link Features

The results of the ablation experiment for hidden link inference are shown in Table 2. We can observe from the experimental results that the features designed for hidden link inference have a significant impact on the accuracy of the inference. The table reveals that among the selected features, the accuracy of inferring hidden links declines most notably upon removal of the number of high-level neighbors while retaining other features, resulting in a 7.15% decrease in inference accuracy. Removal of the AS geographic location leads to the least decline in accuracy, albeit still resulting in a 2.01% decrease in inference accuracy. In general, removing any feature leads to a decrease in method accuracy ranging from 2% to 7%, indicating the effectiveness of both features validated by Toposcope [13] and the newly added features in inferring hidden AS links.

Table 2. Hidden link feature ablation experiment results.

Removed Features	Accuracy Variation
AS Degree	−5.88%
AS Transit Degree	−3.11%
AS Hierarchy	−2.42%
AS Pagerank	−6.04%
AS Organizational Type	−2.21%
AS Customer Cone Size	−4.10%
AS Geographic Location	−2.01%
Number of High-Hierarchy Neighbors	−7.15%

4.5.2. Complex Business Relationships Features

The results of ablation experiments on the inference of complex AS business relationships are presented in Table 3. The experimental results indicate that the features designed for hidden link inference significantly impact the accuracy of the inference. The table illustrates that the accuracy of our method remains relatively stable when removing individual features while keeping others, potentially attributed to the interplay of multiple features in inferring complex business relationships. Nonetheless, the removal of a single feature results in a decrease in inference accuracy, suggesting the continued positive contribution of these features in inferring complex AS business relationships.

Table 3. Ablation experiment results for complex AS business relationship inference features.

Removed Features	Accuracy Variation
AS Degree	−0.52%
AS Transit Degree	−0.14%
Distance to Clique AS	−1.11%
AS Organizational Type	−0.22%
AS Hierarchy	−0.99%
Number of High-Hierarchy Neighbors	−1.33%
AS Pagerank	−0.18%
AS Geographic Location	−0.92%

5. Conclusions

Through the analysis of contemporary internet behavior and inspiration drawn from previous work by internet researchers, we propose Hidden-SAGE, a meticulously designed framework capable of discovering hidden links between Autonomous Systems (AS) and inferring complex business relationships among AS. In comparison with the state-of-the-art algorithm Toposcope, which is capable of discovering hidden links and inferring their business relationships, we find that Hidden-SAGE exhibits better accuracy and fewer

restrictions in discovering hidden links. Additionally, it can infer a greater variety of business relationships. Simultaneously, we conducted ablation experiments on the carefully designed features, obtaining insights into the significance of each feature for our model's performance. Our work indicates that utilizing Hidden-SAGE allows for a more effective expansion of the internet topology, thereby constructing a richer and more comprehensive rich-text internet topology. For future work, our initial priority should be refining our feature design to enhance the accuracy of inferring complex AS business relationships. It is worth noting that our study, as discussed in Section 3.2.1, ignores hybrid types of business relationships when inferring complex business relationships due to the limited availability of training samples. Hence, there is potential for enhancing our methodology to infer hybrid types of business relationships in future endeavors. Additionally, our research has resulted in an enriched topology map of the AS-level Internet and a more detailed delineation of AS business relationships and resources that can be subsequently leveraged to address the route leak problem.

Author Contributions: Conceptualization, H.G., N.L. and Y.X.; Methodology, H.G., N.L. and Y.X.; Software, H.G.; Validation, H.G., N.L. and Y.X.; Formal analysis, H.G. and Y.X.; Investigation, H.G.; Resources, N.L.; Data curation, H.G.; Writing—original draft, H.G.; Writing—review & editing, N.L. and Y.X.; Visualization, H.G.; Supervision, N.L. and Y.X.; Project administration, N.L. and Y.X.; Funding acquisition, N.L. All authors have read and agreed to the published version of the manuscript.

Funding: The research leading to the presented results has been undertaken within the National Natural Science Foundation of China (Youth Project) under Grant No. 62101159, the Natural Science Foundation of Shandong (General Program) under Grant No. ZR2021MF055.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: <https://github.com/GlobalRouteData/Hidden-SAGE> (accessed on 10 April 2024).

Conflicts of Interest: There are no conflicts of interest.

References

1. Zhang, B.; Liu, R.; Massey, D.; Zhang, L. Collecting the Internet AS-level topology. *ACM SIGCOMM Comput. Commun. Rev.* **2023**, *35*, 53–61. [\[CrossRef\]](#)
2. University of Oregon Route Views Project. Available online: <http://www.routeviews.org/routeviews/> (accessed on 19 December 2023).
3. RIPE(RIS). Available online: <https://www.ripe.net/ris/> (accessed on 19 December 2023).
4. Isolario. Available online: <https://www.isolario.it/> (accessed on 23 October 2023).
5. Chang, H.; Govindan, R.; Jamin, S.; Shenker, S.J.; Willinger, W. Towards capturing representative AS-level Internet topologies. *Comput. Netw.* **2004**, *44.6*, 737–755. [\[CrossRef\]](#)
6. Chen, K.; Hu, C.; Zhang, W.; Chen, Y.; Liu, B. On the eyeshots of bgp vantage points. In Proceedings of the GLOBECOM 2009—2009 IEEE Global Telecommunications Conference, Honolulu, HI, USA, 4 March 2010; pp. 1–6.
7. Oliveira, R.; Pei, D.; Willinger, W.; Zhang, B.; Zhang, L. The (in) completeness of the observed Internet AS-level structure. *IEEE/ACM Trans. Netw.* **2009**, *18*, 109–122. [\[CrossRef\]](#)
8. Milolidakis, A.; Bühler, T.; Wang, K.; Chiesa, M.; Vanbever, L.; Vissicchio, S. On the Effectiveness of BGP Hijackers That Evade Public Route Collectors. *IEEE Access* **2023**, *11*, 31092–31124. [\[CrossRef\]](#)
9. Qiu, J.; Gao, L.; Ranjan, S.; Nucci, A. Detecting bogus BGP route information: Going beyond prefix hijacking. In Proceedings of the 2007 Third International Conference on Security and Privacy in Communications Networks and the Workshops-SecureComm 2007, Nice, France, 17–21 September 2007; pp. 381–390.
10. Sermpezis, P.; Kotronis, V.; Dainotti, A.; Dimitropoulos, X. A survey among network operators on BGP prefix hijacking. *ACM SIGCOMM Comput. Commun. Rev.* **2018**, *48*, 64–69. [\[CrossRef\]](#)
11. Problem Definition and Classification of BGP Route Leaks. Available online: <https://www.rfc-editor.org/rfc/rfc7908> (accessed on 23 October 2023).
12. Gregori, E.; Improta, A.; Lenzini, L.; Rossi, L.; Sani, L. A novel methodology to address the internet as-level data incompleteness. *IEEE/ACM Trans. Netw.* **2014**, *23*, 1314–1327. [\[CrossRef\]](#)
13. Jin, Z.; Shi, X.; Yang, Y.; Yin, X.; Wang, Z.; Wu, J. Toposcope: Recover as relationships from fragmentary observations. In Proceedings of the ACM Internet Measurement Conference, Pittsburgh, PA, USA, 27–29 October 2020; pp. 266–280.
14. Gao, L. On inferring autonomous system relationships in the Internet. *IEEE/ACM Trans. Netw.* **2001**, *9*, 733–745.
15. Luckie, M.; Huffaker, B.; Dhamdhere, A.; Giotsas, V.; Claffy, K.C. AS relationships, customer cones, and validation. In Proceedings of the 2013 Conference on Internet Measurement Conference, Barcelona, Spain, 23–25 October 2013; pp. 243–256.

16. Jin, Y.; Scott, C.; Dhamdhere, A.; Giotsas, V.; Krishnamurthy, A.; Shenker, S. Stable and practical AS relationship inference with ProbiLink. In Proceedings of the 16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19), Boston, MA, USA, 26–28 February 2019; pp. 581–598.
17. Peng, S.; Shu, X.; Ruan, Z.; Huang, Z.; Xuan, Q. Classifying multiclass relationships between ASes using graph convolutional network. *Front. Eng. Manag.* **2022**, *9*, 653–667. [\[CrossRef\]](#)
18. Giotsas, V.; Luckie, M.; Huffaker, B.; Claffy, K.C. Inferring complex AS relationships. In Proceedings of the 2014 Conference on Internet Measurement Conference, Vancouver, BC, Canada, 5–7 November 2014; pp. 23–30.
19. Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Philip, S.Y. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 4–24. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Hamilton, W.; Ying, Z.; Leskovec, J. Inductive representation learning on large graphs. in *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Long Beach, CA, USA 2017; Volume 30.
21. Roughan, M.; Willinger, W.; Maennel, O.; Perouli, D.; Bush, R. 10 lessons from 10 years of measuring and modeling the internet's autonomous systems. *IEEE J. Sel. Areas Commun.* **2011**, *29*, 1810–1821. [\[CrossRef\]](#)
22. Mao, Z.M.; Qiu, L.; Wang, J.; Zhang, Y. On AS-level path inference. In Proceedings of the 2005 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, Banff, AB, Canada, 6–10 June 2005; pp. 339–349.
23. Deng, W.; Mühlbauer, W.; Yang, Y.; Zhu, P.; Lu, X.; Plattner, B. Shedding light on the use of AS relationships for path inference. *J. Commun. Netw.* **2012**, *14*, 336–345. [\[CrossRef\]](#)
24. Mühlbauer, W.; Feldmann, A.; Maennel, O.; Roughan, M.; Uhlig, S. Building an AS-topology model that captures route diversity. *JACM SIGCOMM Comput. Commun. Rev.* **2006**, *36*, 195–206. [\[CrossRef\]](#)
25. Mühlbauer, W.; Uhlig, S.; Fu, B.; Meulle, M.; Maennel, O. In search for an appropriate granularity to model routing policies. *ACM SIGCOMM Comput. Commun. Rev.* **2007**, *37*, 145–156. [\[CrossRef\]](#)
26. Dimitropoulos, X.; Krioukov, D.; Fomenkov, M.; Huffaker, B.; Hyun, Y.; Claffy, K.C.; Riley, G. AS relationships: Inference and validation. *ACM SIGCOMM Comput. Commun. Rev.* **2007**, *37*, 29–40. [\[CrossRef\]](#)
27. Faratin, P.; Clark, D.D.; Bauer, S.; Lehr, W.; Gilmore, P.W.; Berger, A. The growing complexity of Internet interconnection. *Commun. Strategies* **2008**, *72*, 51.
28. Dual Transit/Peering. Available online: <http://drpeering.net/white-papers/Art-OfPeering-The-Peering-Playbook.html#4> (accessed on 19 December 2023).
29. Partial Transit (Regional). Available online: <http://drpeering.net/white-papers/Art-OfPeering-The-Peering-Playbook.html#7> (accessed on 19 December 2023).
30. Kastanakis, S.; Giotsas, V.; Livadariu, I.; Suri, N. Replication: 20 Years of Inferring Interdomain Routing Policies. In Proceedings of the 2023 ACM on Internet Measurement Conference, Montreal, QC, Canada, 24–26 October 2023; pp. 16–29.
31. Examine Border Gateway Protocol Case Studies. Available online: <https://www.cisco.com/c/en/us/support/docs/ip/border-gateway-protocol-bgp/26634-bgp-toc.html#toc-hId--910520404> (accessed on 23 October 2023).
32. Jin, W.; Liu, X.; Ma, Y.; Aggarwal, C.; Tang, J. Feature overcorrelation in deep graph neural networks: A new perspective. *arXiv* **2022**, arXiv:2206.07743.
33. CAIDA Public Dataset. Available online: https://publicdata.caida.org/datasets/as-classification_restricted/ (accessed on 23 October 2023).
34. Carmi, S.; Havlin, S.; Kirkpatrick, S.; Shavitt, Y.; Shir, E. A model of Internet topology using k-shell decomposition. *Proc. Natl. Acad. Sci. USA* **2007**, *104*, 11150–11154. [\[CrossRef\]](#)
35. Yang, Y.; Shi, X.; Yin, X.; Wang, Z. The power of prefix hijackings in the internet hierarchy. In Proceedings of the 2016 IEEE Trustcom/BigDataSE/ISPA, Tianjin, China, 23–26 August 2016; pp. 228–235.
36. Wikipedia. Available online: <https://en.wikipedia.org/wiki/PageRank> (accessed on 23 October 2023).
37. Li, J.; Cao, J.; Meng, Z.; Xie, R.; Xu, M. RoLL: Real-Time and Accurate Route Leak Location with AS Triplet Features. In Proceedings of the ICC 2023—IEEE International Conference on Communications, Rome, Italy, 28 May–1 June 2023; pp. 5240–5246.
38. CAIDA Public Dataset. Available online: <https://publicdata.caida.org/datasets/as-organizations/> (accessed on 23 October 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.