China dominates the Rare Earth Industry How do we measure it?

Nils Lang
Waseda University

October 16, 2025

Note: This is a work-in-progress draft, please do not cite or distribute without permission.

1 Introduction

1.1 Background

With industrial policies including but not limited to Export Quota, Production Quota and Environmental Regulations, China achieved a dominance in the Rare Earth industry and its supply chain. This leverage was used in the trade negotiations with the US, after the second Trump administration imposed steeps tariffs on the nation. This was "upending supply chains central to automakers, aerospace manufacturers, semiconductor companies and military contractors around the world." [11] There has already been extensive research into the supply chain of REs, utilizing different approaches to measure Chinas role in the global trade network of RE products. In their paper on the "Evolution of the Rare Earth Trade Network" Xu et al. construct a Dependency Index and use PageRank as one of their network indicators. [13] Hu et al. take a look at a multitude of structural characteristics in their paper "Impacts of China's exports decline in rare earth primary materials from a trade network-based perspective" as well as in-degree centrality and out-degree centrality. [7] Zuo et al. use Network density, Network degree, Betweenness Centrality, Clustering Coefficient and Reciprocity in as the Network feature indicators in their research. [17]
The obvious dominance and usage of it as a foreign policy tool by China offers a unique opportunity to analyze trade networks. By acknowledging the given reality of a highly concentrated market with a known "Hegemon" of sorts allows us to test different approaches of measuring Chinas dominance in the Global Rare Earth Trade Network. Importantly I will not take the approach of "Flattening" the network, because the data inside of China is somewhat approximately corresponding to the Production Quota issued by the XY, but this is not possible to do for all nations.

2 Modeling Global Trade

Complex network theory as a tool to study and model international trade comes originally from analyzing trading communities in the global oil trade. [14] [16] Complex network methods can analyze the world-wide trade system to discover new insights topologically and dynamically. Networks are quite intuitive in the way they model and abstract global trading relations. Originally, complex networks come from analyzing social groups, which can make some algorithm or established indices from the main discipline hard to carry over into global trade network analysis. Different approaches, indices and centralities will be looked at for the trade networks discussed below.

2.1 Network Construction and Capabilities

In order to analyze and visualize the political economy of the rare earth trade, I will break up the collected data into three distinct trade networks corresponding to the below defined capabilities. $TradeNetwork_c$ is built as a directed and weighted network corresponding to the Capability $c \in (1, 2, 3)$:

$$TradeNetwork_c^t = (V^t, E^t, \boldsymbol{W}^t)$$

where

- V^t represents the set of countries participating in the network in year t, serving as the network's **nodes**. Importantly for later on interpretation of Density and different approaches to Centrality, only countries with **at least one trade** for every year t are included in the node set.
- $E^t = \{e_{ij}^t : i, j \in V^t\}$ denotes the set of **directed trade links** (or **edges**) between countries in year t. Each edge e_{ij}^t signifies a trade relationship where trade flows from country j to country i.
- $\boldsymbol{W}^t = \{w_{ij}^t : i, j \in V^t\}$ is the set of **edge weights** for year t. Each weight w_{ij}^t quantifies the **trade value** that country i sends to country j in year t.

2.2 HS Codes

The **Harmonized System (HS) Codes**, established by the World Customs Organization (WCO) and entering into force in 1988, were created to standardize the classification of traded products internationally. This standardized, 6-digit numerical system allows customs authorities, statistical agencies, and governments worldwide to uniformly identify goods for purposes like determining tariffs, monitoring trade flows, and enforcing regulations, significantly streamlining the international trade process. However, the system's inherent limitations arise primarily from the challenge of keeping pace with technological advancements and the emergence of complex, multi-component products (e.g. smartphones or drones) that don't fit neatly into existing categories, leading to classification disputes or ambiguities. Furthermore, while **the initial six digits are globally consistent**, countries often add their own specific digits for national tariff and statistical needs, creating variations (e.g. 10-digit codes in the US and EU) that can still complicate international transactions and necessitate diligence to avoid costly misclassification and customs delays.

Reference	Timeframe	HS Codes	Database	Notes
[13]	2002–2018	Cap1(HS253090), Cap2(HS280530, HS284610, HS284690), Cap3(HS850511)	UN Comtrade	They combine all the Capabilities into one
[7]	1990–2019	HS280530, HS284610, HS284690	UN Comtrade	Mean Between Reporters
[17]	2005–2020	Cap1(HS253090), Cap2(HS280530), Cap3(HS850511)	UN Comtrade	
[4]	2011–2015	HS2846	UN Comtrade	Using HS4
[6]	1996–2015	HS280530	UN Comtrade	Only Import Data
[12]	2002–2014	HS280530	UN Comtrade	Only Import Data
[15]	2001–2010	HS2846, HS85051110	GA Customs China	Chinese HS Codes, Monthly Data
[5]	2010	HS28053010, HS28053090, HS28461000, HS28469000	Eurostat	Only Import Data
[8]	1990–2014	HS280530, HS2846	UN Comtrade	

Table 1: Overview of HS codes, data sources, and methodologies in rare earth trade literature.

UN Comtrade	
HS253090	Mineral substances; n.e.c. in chapter 25
HS280530	Earth-metals, rare; scandium and yttrium, whether or not intermixed or interalloyed
HS2846	Compounds, inorganic or organic, of rare-earth metals; of yttrium or of scandium or of mixtures of these metals
HS284610	Cerium compounds;
HS284690	Compounds, inorganic or organic (excluding cerium), of rare-earth metals, of yttrium, scandium or of mixtures of these metals
HS850511	Magnets; permanent magnets and articles intended to become permanent magnets after magnetisation, of metal
Eurostat	
HS28053010	Intermixtures or interalloys of rare-earth metals, scandium and yttrium
HS28053090	Rare-earth metals, scandium and yttrium, of a purity by weight of <95% (excl. intermixtures and interalloys) (Changed to HS28053040 and HS28053080 in 2016)
HS28461000	Cerium compounds
HS28469000	Compounds of mixtures of rare-earth metals, yttrium and scandium, inorganic or organic (Changed to HS28469030 and HS28469090 in 2016)
GACC	
HS2846	Compounds, inorganic or organic, of rare-earth metals, of yttrium or of scandium or of mixtures of these metals
HS85051110	Permanent magnets of rare-earth metals

Table 2: Descriptions of relevant HS codes.

2.3 Data Description

The basis for the estimation of Chinas dominance over the rare earth industry will be constructed with import and export data collected from the UN Comtrade Database for the years 1989 until 2020. The HS Codes used are the same as Xu et. al use.[13] HS253090 (mineral substances not elsewhere specified), HS284610 (cerium compounds), HS284690 (compounds, inorganic or organic excluding cerium, of rare-earth metals, of yttrium, scandium, or mixtures of these metals), HS280530 (rare-earth metals, scandium and yttrium) and HS850511 (permanent magnets). If not otherwise specified, the used metric will be the amount traded in USD. To construct trade links, I will adopt the data preparation methodology by Hu et al.[7] to address challenges in reported trade data, specifically inconsistencies arising from reporting countries non-compliance with measurement protocols or from label switching during transit for goods sourced from illegal mines. I will define the trade links as exports based on:

$$W_{ij} = \frac{w_{ij}^{\text{export}} + w_{ji}^{\text{import}}}{2}$$

where

- W_{ij} is the trade value that country i exported to country j
- w_{ij} is the amount country i reported as exports to country j
- w_{ji} is the amount country j reported as imports from country i

The UN-published "Practical Guide to Trade Policy Analysis" [1] also mentions BACI[1] as a database, which can be used to interpret trade policies and flows. In their working paper explaining the way they constructed their database, they recommend "BACI is a useful tool for international trade analysis at high degrees of disaggregation, in complement to COMTRADE." [3] It seems like good practice to look at both databases to compare differences in results, while also thinking about the differences in their calculations. BACI only uses algorithms and transformations to the data of UN-Comtrade to clean up the data based on mirroring and other techniques, so in future essays I would like to make a comparison of the two databases. Notably non of the above cited papers use BACI for constructing their networks. In addition to this double data foundation approach, it seems like a good idea in highly concentrated market to also take a look at the

[1]BASE POUR L'ANALYSE DU COMMERCE INTERNATIONAL

Statistics of the individual countries with high importance and compare their published data with the two databases. Looking at country specific Customs Data and the Statistics Yearbooks they have released to test for major inconsistencies.

2.4 Inconsistencies in the Data

In order to remain unbiased an equal weighing of reported imports and exports may seem like the right approach, but the UN also acknowledges in their "Practical Guide to Trade Policy Analysis" [1] that import data is generally more trustworthy. Since Customs Offices are generally there to collect taxes based on tariffs the data they collect are intuitively better. Relying only on import data may on the other hand lead to a bias towards countries who have a lot of tariffs or better functioning customs offices. Xu et al. [13] take the approach of using all the import data and only taking the export data, if the trade was not reported by the importer. Even when using the data from BACI, the data bases only covers the trade after 1995 and therefore this should be applied for data from before.
I suggest a dynamic approach based on previous inconsistencies in the reported data over the to be analyzed timeframe. This can either be done using all the data accessible at the corresponding HS-Level or only for the specific commodity over the to be looked at timeframe. Since there was a lot of illegal mining and incentives to misreport exports in the rare earth industry not present in the entire global trade network, I would like to take the second approach.
In a first step I define **Inconsistencies in Trade**: While there will always be inconsistencies in data, trade data is specifically prone to being misreported. Roots could be incentives to avoid tariffs, circumvent trade restriction or simple measurement errors by the customs offices countries. [9] In order to install more confidence into the data, I will for now take a look at the inconsistencies over time. The measurement of the inconsistencies will be done as follows: *inconsistency* as

$$\Delta_{ij} = \left| w_{ij}^{\text{export}} - w_{ji}^{\text{import}} \right|$$

For each $TradeNetwork_c$ the inconsistencies will then be normalized to make them comparable over time and across capabilities using a Min-Max scaling Formula and can be seen visualized in Figure 1:

$$\Delta_{ij}^{Normalized} = \frac{\Delta_{ij} - \operatorname{argmin} \Delta_c}{\operatorname{argmax} \Delta_c - \operatorname{argmin} \Delta_c}$$

where

- Δ_{ij} is the difference in reported values by i and j for the goods going into j
- $\operatorname{argmin} \Delta_c$ is the smallest value of Δ_{ij} in $TradeNetwork_c$
- $\operatorname{argmax} \Delta_c$ is the largest value of Δ_{ij} in $TradeNetwork_c$

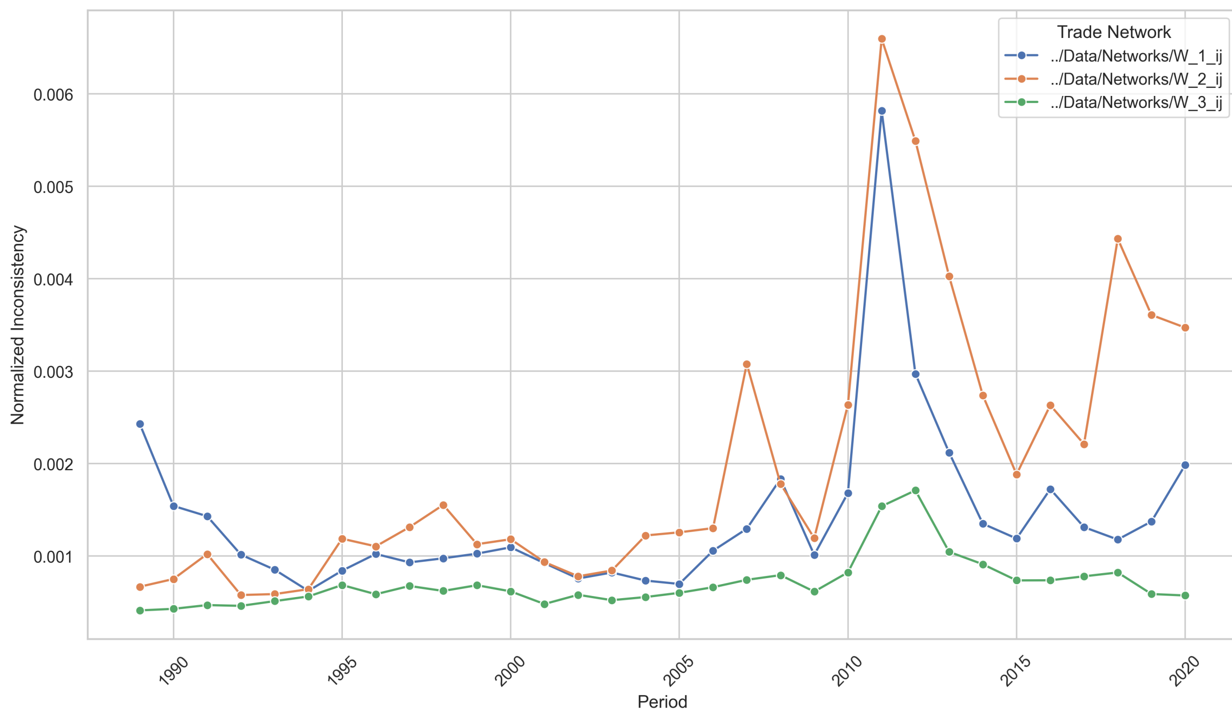

Figure 1: Inconsistencies in the Trade Networks

3 Analysis of Trade Networks

The goal of analyzing Trade Networks is to be able to point out crucial connections and find important or "central" nodes within the network. Over the time there have been multiple approaches by different disciplnes to use networks and their topological features to further knowledge. Originally based in Computer Science, Network Analysis has since become a popular tool for analyzing (especially big) social networks.

3.1 PageRank Centrality

Let N be the number of nodes in $TradeNetwork_c$, and let $\boldsymbol{W}$ be the weighted adjacency matrix such that W_{ij} represents the weight of the edge from node i to node j. Let R be the vector storing scores r_i with $i \in \{1, \ldots, N\}$, with initial values

$$R = \begin{bmatrix} \frac{1}{N} \\ \frac{1}{N} \\ \vdots \\ \frac{1}{N} \end{bmatrix} \in \mathbb{R}^N.$$

The PageRank score r_i for node i is defined recursively as:

$$r_i = \alpha \sum_{j=1}^{N} \frac{W_{ji}}{\sum_{k=1}^{N} W_{jk}} r_j + (1 - \alpha) \frac{1}{n},$$

where:

- r_i is the PageRank score of node i,
- α is the damping factor ($\alpha = 0.85$),

- W_{ij} is the weight of the edge from node i to node j,
- $\sum_{k=1}^{n} W_{kj}$ is the total outgoing weight from node j,
- N is the total number of nodes in the network.

The original paper for the PageRank algorithm was written by Lawrence Page in 1998. [10] While there was some search engines out there already none of them would be a successful as the one founded on this algorithm - today we know it under the name Google.
Page et al. came up with the concept of ranking the importance of a website by the amount of important websites pointing towards it.

In the Appendix my calculation process is explained more in detail, but the evolution of Chinas Page Rank can be seen below.

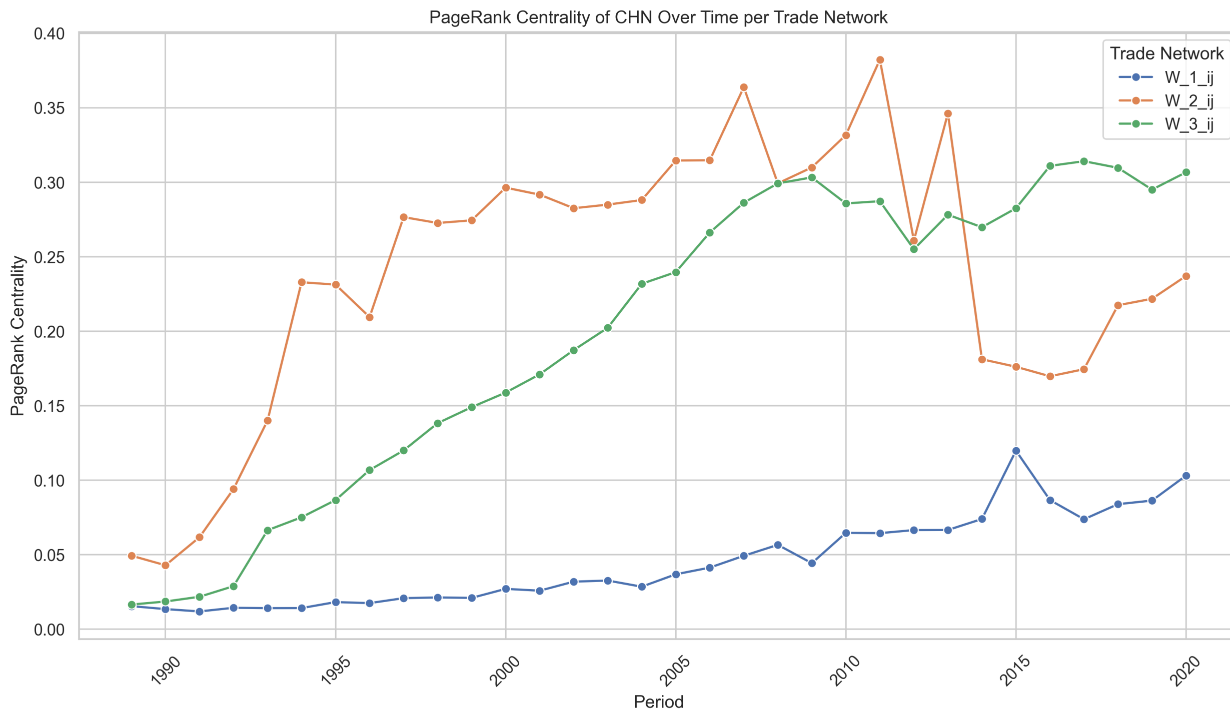

Figure 2: PageRank Centrality for China

The problem with PageRank

The way the original PageRank is constructed over-represents countries who imports a lot of goods and therefore is not well suited for analyzing market power as can be seen in Figure 3 for Thailand and Japan. The exact visualization technique is explained in the Appendix as well, but in short the Nodesize is the matrix-calculated PageRank as seen above and the edges are the weight of the trade flow. Possible solutions to better measure Market Power could be:

- Bidirectional PageRank looking also at exports and then dividing by two
- Looking how Networks would react, if an actor would disappear.
- Betweenness Centrality which allows for $C + 1$ to be next step in the path. Could be a good model of real life supply chains.

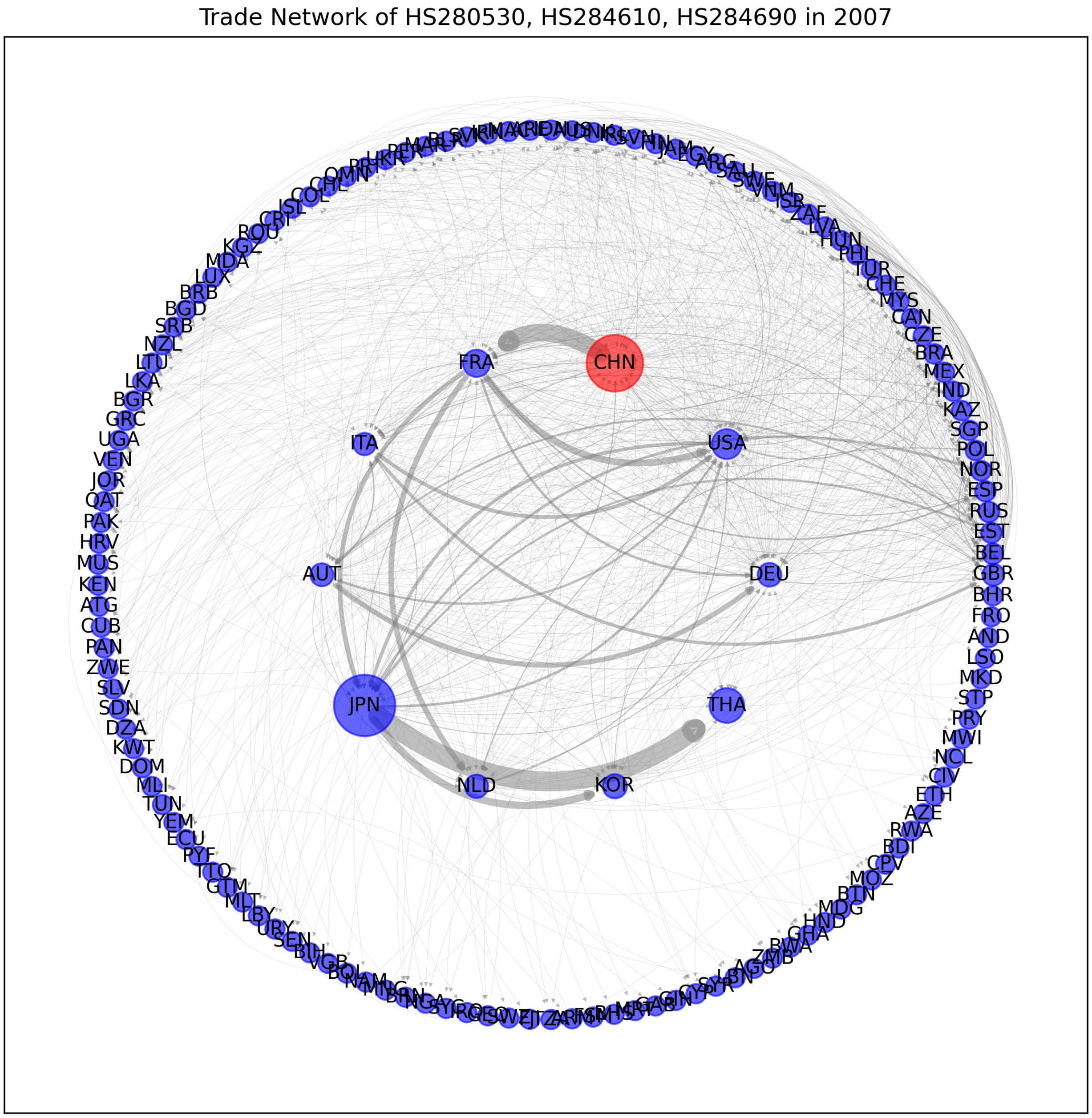

Figure 3: TradeNetwork for Cap 2 in 2007

3.2 Density (D)

Density measures the overall connectedness of the network by comparing the number of existing edges to the maximum possible number of edges.

$$D = \frac{E}{N(N-1)}$$

where E is the number of edges in network $TradeNetwork_c^t$ and N is the number of nodes in the network. The density D ranges from 0 to 1. A higher value indicates tighter connections and a more "complete" network.

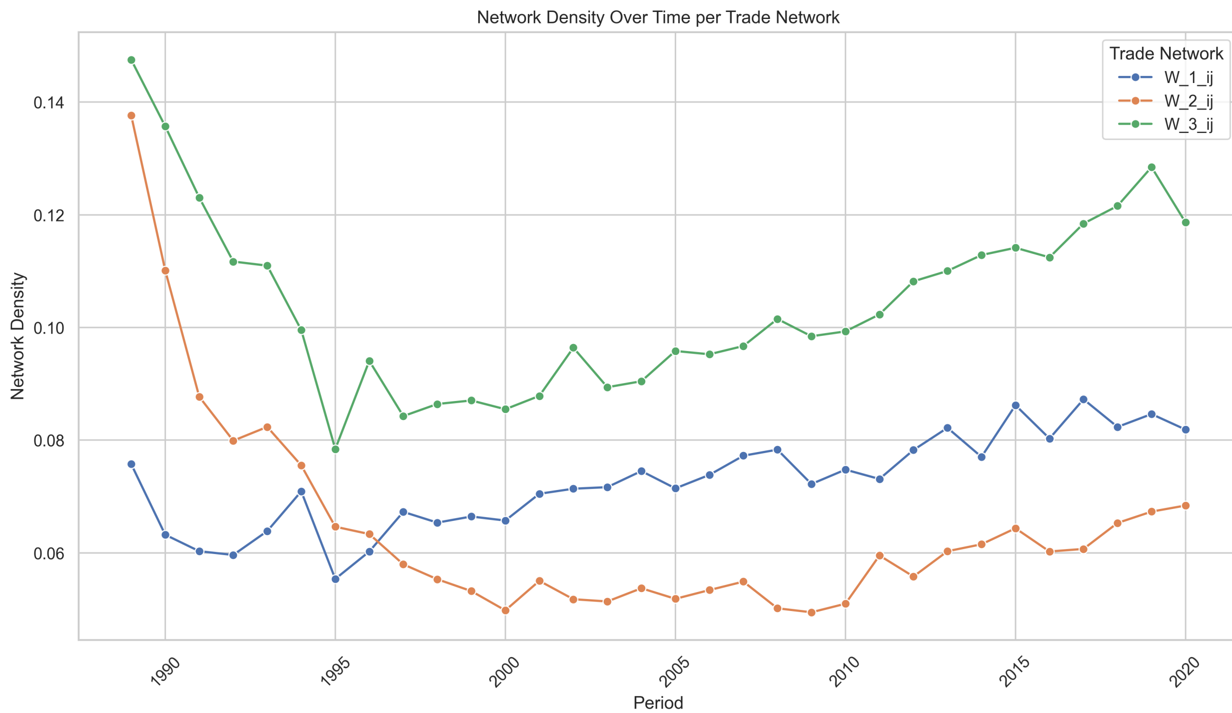

Figure 4: Density of the Networks per year and capability

3.3 In-degree Centrality (C_I^{in})

This measures the number of incoming connections for a country. In the context of import trade, a high in-degree centrality for a country suggests it imports from many other countries, indicating a diverse range of trade partners. The in-degree centrality of a specific country i, denoted as $C_I^{in}(i)$, is simply **the count of incoming edges** to that country's node.

$$C_I^{in}(i) = k_i^{in}$$

For better comparison across different networks, we can use the normalized in-degree centrality. This metric scales the in-degree centrality to a value between 0 and 1 by dividing it by the maximum possible in-degree, which is one less than the total number of nodes ($N-1$).

$$C_I^{in}(i)_{\text{normalized}} = \frac{k_i^{in}}{N-1}$$

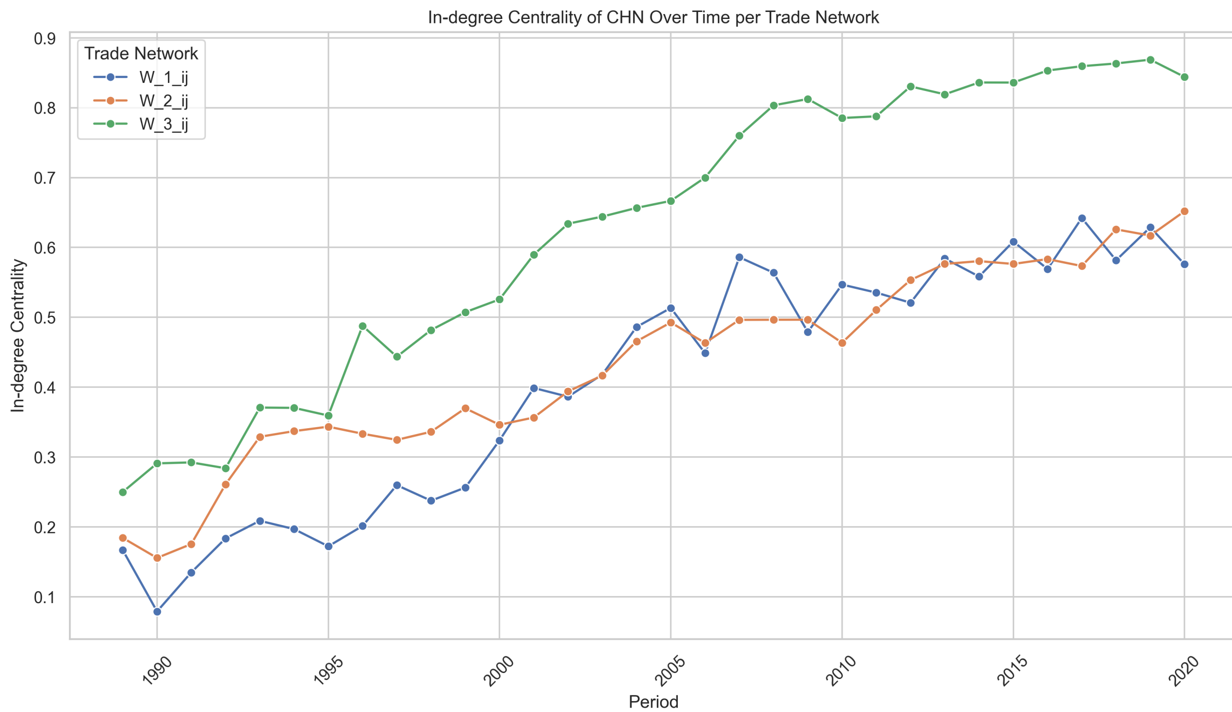

Figure 5: In-degree Centrality for China

3.4 Out-degree Centrality (C_I^{out})

This measures the number of outgoing connections for a country. In the context of export trade, a high out-degree centrality for a country suggests it exports to many other countries, indicating a diverse range of trade partners. The out-degree centrality of a specific country i, denoted as $C_O^{out}(i)$, is simply the **count of outgoing edges** from that country's node.

$$C_O^{out}(i) = k_i^{out}$$

For better comparison across different networks, we can use the normalized out-degree centrality. This metric scales the out-degree centrality to a value between 0 and 1 by dividing it by the maximum possible out-degree, which is one less than the total number of nodes ($N - 1$).

$$C_O^{out}(i)_{\text{normalized}} = \frac{k_i^{out}}{N - 1}$$

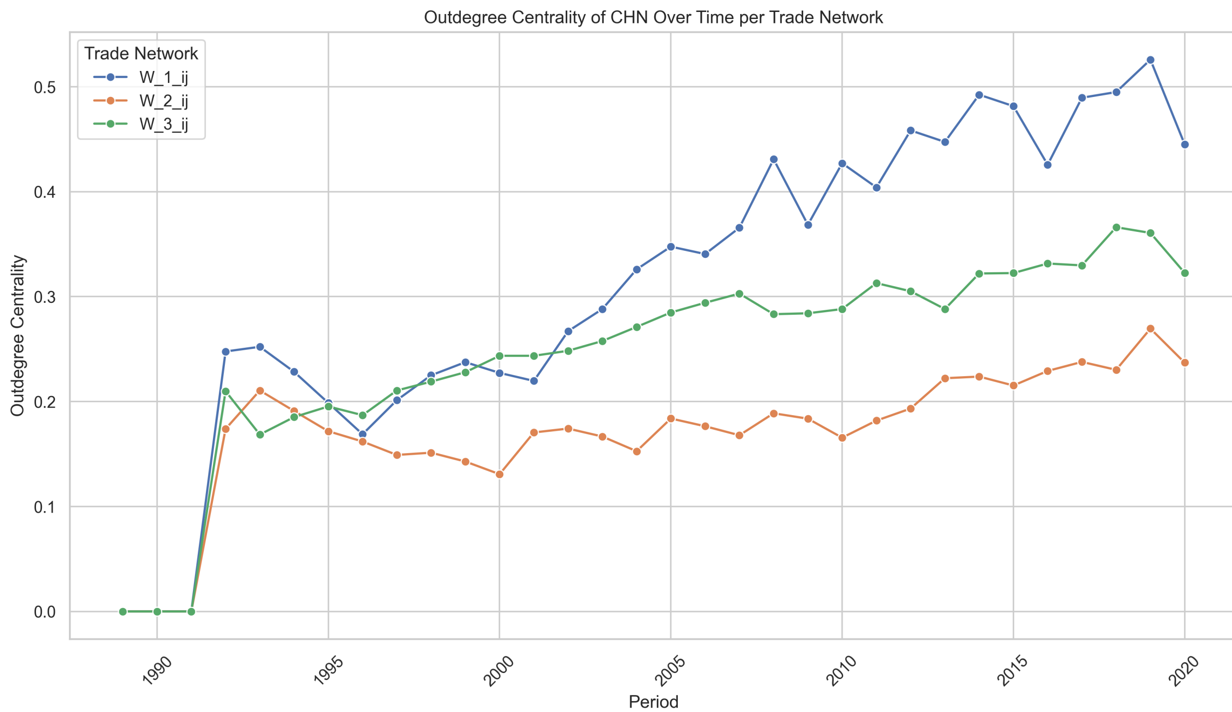

Figure 6: Out-degree Centrality for China

3.5 Degree Centrality (C^D)

Degree centrality is a fundamental measure of a node's influence or activity within a network. It is calculated as the sum of a node's incoming and outgoing connections. In the context of international trade, a country with high degree centrality is a significant player, acting as both a major importer and exporter.

The degree centrality of a specific country i, denoted as $C^D(i)$, is the sum of its in-degree (k_i^{in}) and out-degree (k_i^{out}).

$$C^D(i) = k_i^{in} + k_i^{out}$$

For better comparison across different networks, we can use the normalized degree centrality. This metric scales the degree centrality to a value between 0 and 2.

$$C^D(i)_{\text{normalized}} = \frac{k_i^{in} + k_i^{out}}{(N-1)}$$

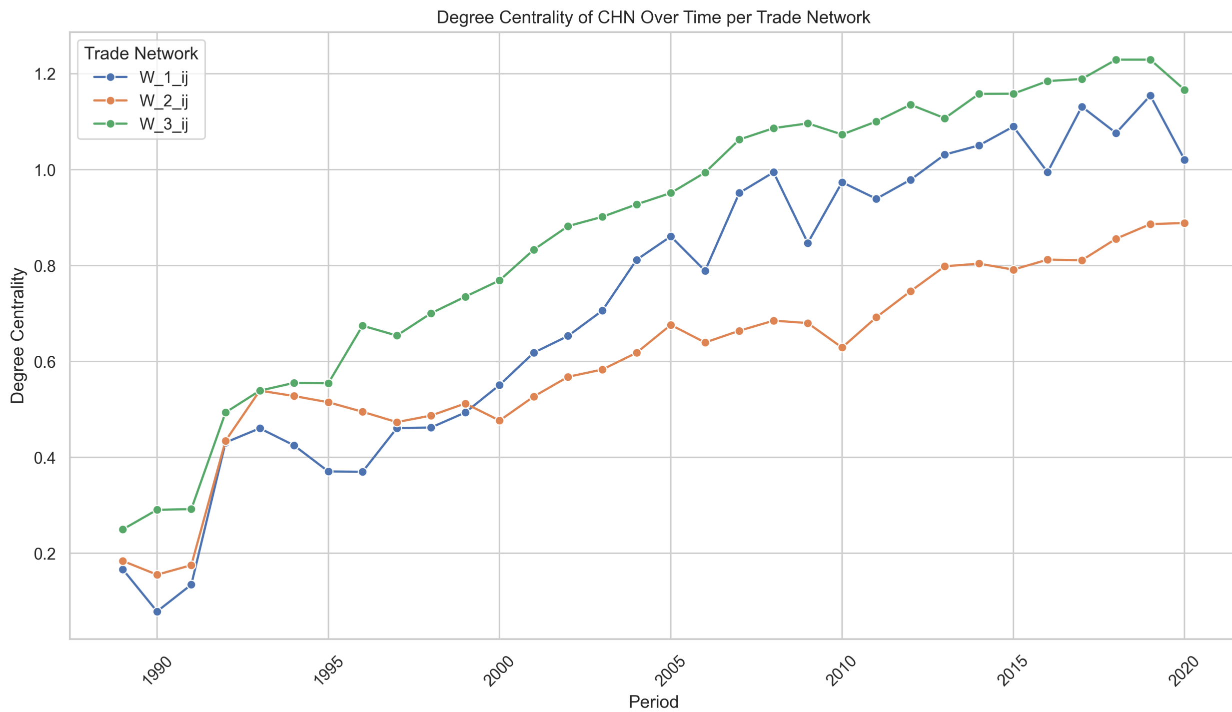

Figure 7: Degree Centrality for China

4 Economic Interpretation of Results

Since only countries involved in the Rare Earth Trade are present in the networks, there is selection bias present, when it comes to the interpretation of degree centralities. The insights gain from Figure 5 are still quite profound, with China importing from close to 84% of all countries who export permanent magnets. The numbers for raw materials and metals are both close to 60%, which is also notable. This seems to indicate that there is at least some form of reliance on foreign imports for domestic chinese importers, when it comes to the RE Industry. It should be noted though, that this is a **non-deflated network** meaning the supply chain in China itself is not displayed. This could seriously dilute the data, since China has the biggest production of all mentioned commodities along the supply chain.
The general increase in C^D can be viewed as China breaking into new markets for their products to sell into. The density also increases after 2000, which shows the slow process of an emerging globalized market. The falling density in the 1990s can be interpreted as former soviet countries joining global trade and consequently decreasing density and only slowly building up global connections.
The spike in inconsistencies seen in 2011 for the refined metals is likely explained by an incident for Senkaku/-Diaoyu Islands and the resulting first time use of Rare Earths to pressure Japan to release a fisherman imprisoned there for illegal fishing.

5 Topological Metrics

The following is a rewrite of Freemans 1978 with some slight additions from Hu et al. "Impacts of China's exports decline in rare earth primary materials from a trade network-based perspective" [2] [7] This will allow me to harmonize the declarations of different metrics. Large parts of this will be pushed into the Appendix at some point. For thinking trough it, I will leave it here for now.
Let $TradeNetwork_c^t$ be a graph with N number of nodes and let the connections be represented by a matrix W. **Notably this excludes self exports or self imports.** With i being a node in $TradeNetwork_c^t$ the

following definitions apply:

5.1 In-Degree (k_i^{in})

The in-degree of node i is the number of incoming edges to node i.**TODO Discuss if t is needed for the definitions here, gonna drop it for now**

$$k_i^{in} = \sum_{j=1}^{N} \text{sgn}(W_{ji})$$

where $\text{sgn}(x)$ is 1 if $x > 0$ and 0 otherwise. In a trade network, k_i^{in} represents the number of countries that export to country i.

5.2 Out-Degree (k_i^{out})

The out-degree of node i is the number of outgoing edges from node i.

$$k_i^{out} = \sum_{j=1}^{N} \text{sgn}(W_{ij})$$

In a trade network, k_i^{out} represents the number of countries to which country i exports.

5.3 Total Node Degree (k_i)

The total node degree of node i is the sum of its in-degree and out-degree.

$$k_i = k_i^{in} + k_i^{out}$$

This indicates the total number of distinct trading partners (importers and exporters) for country i. This measure is dependent on the size of the network.

5.4 In-Strength (s_i^{in})

The in-strength of a node i measures the sum of the weights of all incoming connections to node i for period t of the network $\mathit{TradeNetwork}_c^t$. It reflects how much influence or information node i receives from other nodes. In a trade network, s_i^{in} quantifies the total value or volume of imports received by country i at time t.

$$s_i^{in} = \sum_{j=1}^{N} W_{ji}$$

5.5 Out-Strength (s_i^{out})

The out-strength of a node i measures the sum of the weights of all outgoing connections from node i for period t of the network $\mathit{TradeNetwork}_c^t$. It reflects how much influence node i sends to other nodes. In a trade network, s_i^{out} quantifies the total value or volume of exports sent by country i at time t.

$$s_i^{out} = \sum_{j=1}^{N} W_{ij}$$

5.6 Total Node Strength (s_i)

The total activity of a node, which is the sum of its incoming and outgoing weighted connections, can be written as:

$$s_i = s_i^{in} + s_i^{out} = \sum_{j=1}^{N} W_{ji} + \sum_{j=1}^{N} W_{ij}$$

This represents the total weighted trade activity (imports + exports) of country i at time t.

6 Measures of Centrality

6.1 Relative Point Centrality ($C_D(i)$)

To compare the relative centrality of points from different graphs, a measure independent of network size is needed. In Freemans original work he did not account for directed graphs, therefore we redefine the maximum possible degree for any point in a network of n points as not being $n-1$, but $2(n-1)$. Since there can be either exports or imports for any given i in $TradeNetwork_c^t$. The following measure normalizes the degree by its maximum possible value.

$$C'_D(i) = \frac{k_i}{2(n-1)}$$

7 TODO Go over all centralities starting from here and make them consistent, also read Freeman 1977 again

7.1 Network Centralization (C_D)

This index reflects the tendency of a single point to be more central than all others in the network. It is a ratio of the observed sum of differences in point centrality to the maximum possible sum for a network of that size. The maximum difference sum occurs in a star or wheel graph.

$$C_D = \frac{\sum_{i=1}^{n}[C_D(p^*) - C_D(p_i)]}{n^2 - 3n + 2}$$

where p^* is the point with the highest degree centrality.

7.2 Betweenness Centrality

Betweenness centrality is based on the frequency with which a point falls on the shortest paths (geodesics) connecting other pairs of points. An actor in such a position has the potential to influence or control communication by acting as an intermediary.

7.3 Absolute Point Centrality ($C_B(p_k)$)

This measure is a sum of the point's partial betweenness values for all pairs of other points. Partial betweenness is the probability that a point falls on a randomly selected geodesic between two others.

$$C_B(p_k) = \sum_{i<j}^{n}\sum_{j}^{n} b_{ij}(p_k)$$

where $b_{ij}(p_k) = \frac{g_{ij}(p_k)}{g_{ij}}$, with:

- g_{ij} = the number of geodesics linking p_i and p_j.
- $g_{ij}(p_k)$ = the number of geodesics linking p_i and p_j that contain p_k.

7.4 Relative Point Centrality ($C'_B(p_k)$)

To compare betweenness centrality across graphs of differing sizes, a relative measure is used. The maximum possible value for $C_B(p_k)$ is achieved by the central point in a star graph and is equal to $\frac{n^2-3n+2}{2}$.

$$C'_B(p_k) = \frac{2C_B(p_k)}{n^2 - 3n + 2}$$

7.5 Network Centralization (C_B)

This index measures the overall centralization of a network based on betweenness. It is a ratio of the observed sum of differences to the maximum possible sum, which is yielded by the star or wheel graph.

$$C_B = \frac{\sum_{i=1}^{n}[C_B(p^*) - C_B(p_i)]}{(n-1)(n^2-3n+2)}$$

7.6 Closeness Centrality

Closeness centrality is based on the idea that a point is central if it is close to all other points in the graph. This concept is related to a point's independence, as a central point can avoid the need for intermediaries to relay messages. It is also related to efficiency, as shorter distances mean lower costs and faster communication.

7.7 Absolute Point Centrality (as an inverse) ($C_C(p_k)^{-1}$)

As proposed by Sabidussi (1966), this measure is the sum of geodesic distances from a point p_k to all other points in the network. Since the value increases as points are farther apart, it is a measure of "decentrality".

$$C_C(\mathfrak{p}_k)^{-1} = \sum_{i=1}^{n} d(\mathfrak{p}_i, \mathfrak{p}_k)$$

where $d(p_i, p_k)$ is the number of edges in the geodesic linking p_i and p_k. This measure is only meaningful for connected graphs.

7.8 Relative Point Centrality ($C'_C(p_k)$)

To provide a direct measure of closeness that is comparable across networks, Beauchamp's (1965) approach is used. It is defined as the inverse of the average distance from a point to all others.

$$C'_C(p_k) = \frac{n-1}{\sum_{i=1}^{n} d(p_i, p_k)}$$

A point that is maximally close to all others will have a value of unity.

7.9 Network Centralization (C_C)

Similar to the other network measures, this index is based on the differences in closeness centrality between the most central point and all other points in the network. It is also highest for a star or wheel graph.

$$C_C = \frac{\sum_{i=1}^{n}[C'_C(p^*) - C'_C(p_i)]}{(n^2-3n+2)(2n-3)}$$

7.10 Betweenness Centrality

In simple terms, a node's betweenness centrality is calculated by:

Finding all the shortest paths between every pair of nodes in the network. For each node, counting how many of these shortest paths pass through it. Dividing that count by the total number of shortest paths between all pairs of nodes

7.11 Eigenvector Centrality

Measures like *degree centrality* provide a basic understanding, but sometimes fall short in capturing the true influence in a network. A node connected to many other highly influential nodes is arguably more important than a node connected to many isolated or less influential nodes. This is where *Eigenvector Centrality* can help alleviate some concerns. Eigenvector Centrality is a measure of the influence of a node in a network. It assigns relative scores to all nodes based on the principle that connections to high-scoring nodes contribute more to the score of the node in question than equal connections to low-scoring nodes. In essence, it's about "having influential connections."

Consider a network (or graph) with n nodes

The structure of the network can be represented by an *adjacency matrix*, denoted as A.

For an undirected graph, A is an $n \times n$ symmetric matrix where:

$$A_{ij} = \begin{cases} 1 & \text{if node } i \text{ is connected to node } j \\ 0 & \text{otherwise} \end{cases}$$

For a **directed graph**, $A_{ij} = 1$ if there is a link from node j to node i (meaning j influences i). In this case, the matrix A is generally not symmetric.

For a **weighted graph**, the entries A_{ij} represent the *strength* or *weight* of the connection between node i and node j. For instance, if $A_{ij} = 5$, it means the connection between i and j is five times stronger than a connection with a weight of 1. If there is no connection, $A_{ij} = 0$.

$$A_{ij} = \text{weight of connection from node } j \text{ to node } i$$

In the context of eigenvector centrality, a higher weight from an influential node contributes more significantly to the receiving node's centrality score.

7.12 Recursive Definition

Let x_i be the eigenvector centrality score of node i. The core idea of eigenvector centrality is that a node's score is proportional to the sum of the scores of its neighbors, weighted by the strength of the connections. This can be expressed as:

$$x_i = \frac{1}{\lambda} \sum_{j \in N(i)} A_{ij} x_j$$

where $N(i)$ is the set of neighbors of node i (or nodes that link to i in a directed graph), A_{ij} is the weight of the connection from node j to node i, and λ is a constant (a scalar factor).

8 Network Metrics

Let $\mathit{TradeNetwork}_c^t$ be a graph with N number of nodes and let the connections be represented by a matrix W. With i being a node in $\mathit{TradeNetwork}_c^t$ the following metrics can be calculated for a trade network:

8.1 Degree distribution

The degree distribution $P(k)$ of a network is defined to be the fraction of nodes in the network with degree k. The simplest network model, for example, the (Erdős–Rényi model) random graph, in which each of n nodes is independently connected (or not) with probability p (or $1-p$), has a binomial distribution of degrees k (or Poisson in the limit of large n). Most real networks have a degree distribution that is highly right-skewed, meaning that a large majority of nodes have low degree but a small number, known as "hubs", have high degree. For such scale-free networks the degree distribution approximately follows a power law:

$$P(k) \sim k^{-\gamma}$$

where γ is the degree exponent, and is a constant. Such scale-free networks have unexpected structural and dynamical properties, rooted in the diverging second moment of the degree distribution.

8.2 Average Clustering Coefficient (ACC)

The average clustering coefficient measures the tightness of the network by considering the prevalence of "triangles" (closed loops of three nodes) and the weights of edges. For a node i, its clustering coefficient C_i is defined as:

$$C_i = \frac{T_i}{k_i(k_i^{rec} - 1)}$$

where T_i is the number of directed triangles through node i in network $TradeNetwork_c^t$. k_i is the total node degree of node i. k_i^{rec} is the reciprocal degree of node i, defined as the number of neighbors with whom node i has a reciprocal connection (both imports and exports):

$$k_i^{rec} = \sum_{j=1}^{N} \text{sgn}(W_{ij} W_{ji})$$

The average clustering coefficient for the network G_t is then the average of individual node clustering coefficients:

$$ACC = \frac{1}{N} \sum_{i=1}^{N} C_i$$

A higher ACC suggests that countries that trade with a common partner are also more likely to trade directly with each other, indicating the presence of tightly-knit trading groups.

8.3 Heterogeneity (H)

Heterogeneity quantifies the differences in node degrees across the network. It evaluates how unevenly distributed the connections are among nodes.

$$H = \frac{\sqrt{\sum_{i=1}^{N} (k_i - \bar{k})^2 / N}}{\bar{k}}$$

where $\bar{k}$ is the average node degree in network G_t. A high h indicates that a few nodes (countries) have a disproportionately large number of connections, while many others have very few, suggesting a centralized or "hub-and-spoke" structure in the trade network.

9 Measuring Market Power

To understand the development of Chinas market power in the global rare earth trade, I will use a number of indicators to both analytically and visually allow the reader to understand the progressive concentration of power. **Discuss Centrality versus Market Power here**

9.1 Herfindahl-Hirschman Index

The Herfindahl–Hirschman Index (HHI) measures of the size of an entity i in relation to the market they are in and is an indicator of the amount of competition among them. A higher HHI indicates a more concentrated market and thus higher potential for market power. The formula for the HHI is:

$$\text{HHI} = \sum_{i=1}^{N} (S_i)^2 \quad (1)$$

Where:

- S_i is the volume market share of country i.
- N is the number of countries in the market.

10 Comparing Metrics and Indices

11 Visualizing Trade

Representing the flow of trade in an intuitive way is necessary to understand the developments and trends that shaped the rare earth industry over the last decades. My approach to visualize the global trade is heavily influenced by the research done by Xu et al. [13]. In every year the countries at the center of the visualization will be the ten highest ranked according to their TradeRank. I define TradeRank as follows:

$$R = \begin{bmatrix} \frac{1}{N} \\ \frac{1}{N} \\ \vdots \\ \frac{1}{N} \end{bmatrix} \in \mathbb{R}^N.$$

The PageRank score for import importance r_i^{import} for node i is defined recursively as:

$$r_i^{import} = \alpha \sum_{j=1}^{N} \frac{W_{ji}}{\sum_{k=1}^{N} W_{jk}} r_j + (1-\alpha)\frac{1}{n}$$

The PageRank score for export importance r_i^{export} for node i is defined recursively as: where:

$$r_i^{export} = \alpha \sum_{j=1}^{N} \frac{W_{ij}}{\sum_{k=1}^{N} W_{ik}} r_j + (1-\alpha)\frac{1}{n}$$

I then define the TradeRank as being a combination of export and import importance:

$$tr_i = \frac{r_i^{import} + r_i^{export}}{2}$$

References

[1] Marc Bacchetta, World Trade Organization, and United Nations Conference on Trade and Development, eds. *A Practical Guide to Trade Policy Analysis*. Geneva: World Trade Organization : United Nations Conference on Trade and Development, 2012. ISBN: 978-92-870-3812-8.

[2] Linton C. Freeman. "Centrality in Social Networks Conceptual Clarification". In: *Social Networks* 1.3 (Jan. 1978), pp. 215–239. ISSN: 0378-8733. DOI: 10.1016/0378-8733(78)90021-7. (Visited on 07/17/2025).

[3] Guillaume Gaulier and Soledad Zignago. "BACI: International Trade Database at the Product-Level (the 1994-2007 Version)". In: *SSRN Electronic Journal* (2010). ISSN: 1556-5068. DOI: 10.2139/ssrn.1994500. (Visited on 07/06/2025).

[4] Jianping Ge et al. "World Rare Earths Trade Network: Patterns, Relations and Role Characteristics". In: *Resources Policy* 50 (Dec. 2016), pp. 119–130. ISSN: 03014207. DOI: 10.1016/j.resourpol.2016.09.002. (Visited on 08/06/2025).

[5] Dominique Guyonnet et al. "Material Flow Analysis Applied to Rare Earth Elements in Europe". In: *Journal of Cleaner Production* 107 (Nov. 2015), pp. 215–228. ISSN: 09596526. DOI: 10.1016/j.jclepro.2015.04.123. (Visited on 05/22/2025).

[6] Wenyu Hou et al. "Structure and Patterns of the International Rare Earths Trade: A Complex Network Analysis". In: *Resources Policy* 55 (Mar. 2018), pp. 133–142. ISSN: 03014207. DOI: 10.1016/j.resourpol.2017.11.008. (Visited on 08/06/2025).

[7] Xiaoqian Hu et al. "Impacts of China's Exports Decline in Rare Earth Primary Materials from a Trade Network-Based Perspective". In: *Resources Policy* 81 (Mar. 2023), p. 103321. ISSN: 03014207. DOI: 10.1016/j.resourpol.2023.103321. (Visited on 05/22/2025).

[8] Nabeel A. Mancheri. "World Trade in Rare Earths, Chinese Export Restrictions, and Implications". In: *Resources Policy* 46 (Dec. 2015), pp. 262–271. ISSN: 03014207. DOI: 10.1016/j.resourpol.2015.10.009. (Visited on 06/02/2025).

[9] Muryawan Markie. *Understanding Trade Data*. https://www.biological-arms-control.org/projects_tradememonitoring/MSP %20Markie%20Muryawan%20-%20Understanding%20trade%20data.pdf. Dec. 2012. (Visited on 06/24/2025).

[10] Lawrence Page et al. *Page_Rank_OG*. Nov. 1999.

[11] *US Says Deal with Beijing Will Expedite Rare Earth Exports from China — Reuters*. https://www.reuters.com/world/chi says-deal-related-trade-was-signed-with-china-wednesday-2025-06-26/. (Visited on 07/17/2025).

[12] Xibo Wang et al. "Spatial Dynamics of the Communities and the Role of Major Countries in the International Rare Earths Trade: A Complex Network Analysis". In: *PLOS ONE* 11.5 (May 2016). Ed. by Gui-Quan Sun, e0154575. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0154575. (Visited on 08/06/2025).

[13] Jilan Xu et al. "Evolution of the Rare Earth Trade Network: A Perspective of Dependency and Competition". In: *Geoscience Frontiers* 15.3 (May 2024), p. 101653. ISSN: 16749871. DOI: 10.1016/j.gsf.2023.101653. (Visited on 06/25/2025).

[14] Hai-Ying Zhang, Qiang Ji, and Ying Fan. "Competition, Transmission and Pattern Evolution: A Network Analysis of Global Oil Trade". In: *Energy Policy* 73 (Oct. 2014), pp. 312–322. ISSN: 03014215. DOI: 10.1016/j.enpol.2014.06.020. (Visited on 08/06/2025).

[15] Lu Zhang et al. "Did Chinas Rare Earth Export Policies Work? Empirical Evidence from USA and Japan". In: *Resources Policy* 43 (Mar. 2015), pp. 82–90. ISSN: 03014207. (Visited on 06/02/2025).

[16] Weiqiong Zhong et al. "The Evolution of Communities in the International Oil Trade Network". In: *Physica A: Statistical Mechanics and its Applications* 413 (Nov. 2014), pp. 42–52. ISSN: 03784371. DOI: 10.1016/j.physa.2014.06.055. (Visited on 08/06/2025).

[17] Zhili Zuo et al. "Evolution and Insights into the Network and Pattern of the Rare Earths Trade from an Industry Chain Perspective". In: *Resources Policy* 78 (Sept. 2022), p. 102912. ISSN: 0301-4207. DOI: 10.1016/j.resourpol.2022.102912. (Visited on 06/25/2025).

12 Appendix

12.1 Calculating PageRank

The PageRank algorithm, developed by Larry Page and Sergey Brin, is a fundamental method for measuring the importance or authority of nodes within a network, particularly web pages. It interprets links as "votes" where a link from an important page carries more weight. The algorithm uses an iterative process to assign a numerical weight to each element of a hyperlinked set of documents, aiming to establish their relative importance.

1. Data Extraction and Graph Setup

The first step involves preparing the input data, which represents a directed weighted graph. An input DataFrame, `TN_t_c`, contains columns for `reporterISO` (source country), `partnerISO` (destination country), and `W_ij` (the weight of the link). Let N be the total number of unique countries (nodes) in our network. Each country is assigned a unique integer index from 0 to $N-1$.

2. Initializing and Populating the Weighted Adjacency Matrix W

I construct an $N \times N$ weighted adjacency matrix, $\mathbf{W}$.

- **Definition:** An element W_{ij} in this matrix represents the total weight of direct links *from* country i *to* country j. If there is no link from i to j, then $W_{ij} = 0$.
- **Mathematical Representation:** For a set of N countries, the matrix $\mathbf{W}$ is:

$$\mathbf{W} = \begin{pmatrix} W_{00} & W_{01} & \cdots & W_{0,N-1} \\ W_{10} & W_{11} & \cdots & W_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ W_{N-1,0} & W_{N-1,1} & \cdots & W_{N-1,N-1} \end{pmatrix}$$

Each W_{ij} is the sum of weights for all links from country i to country j. With the data structure of different periods this does not matter for my case, but in Capability 1 this leads to me taking the sum of the commodity codes as the weight and not looking at them individually.

- **Conceptual Visualization:** Imagine N countries. If country i links to country j with a weight w, then $W_{ij} = w$. For example, a simplified $\mathbf{W}$ for $N = 4$ countries might look like:

$$\mathbf{W} = \begin{pmatrix} 0 & W_{01} & W_{02} & W_{03} \\ W_{10} & 0 & W_{12} & W_{13} \\ W_{20} & W_{21} & 0 & W_{23} \\ W_{30} & W_{31} & W_{32} & 0 \end{pmatrix}$$

(where $W_{ii} = 0$ for no self-loops unless otherwise specified by data).

3. Creating the Transition Probability Matrix P

The PageRank power iteration formula requires a **column-stochastic matrix P**, where P_{ji} represents the probability of a random surfer moving from page i to page j.

- **a. Calculate Row Sums (Out-Degrees):** For each country i, I calculate the total "strength" of its outgoing links. This is the sum of the i-th row in $\mathbf{W}$.

$$L(i) = \sum_{k=0}^{N-1} W_{ik}$$

This forms a vector $\mathbf{L}$ of total outgoing link weights for each node. L represents the sum of all outgoing connections. If a country does not export at all it is set to 0. This leaves us with:

$$\mathbf{L} = \begin{pmatrix} L(0) \\ L(1) \\ \vdots \\ L(N-1) \end{pmatrix}$$

- **b. Handle Dangling Nodes:** A "dangling node" is a country (node) that has no outgoing links (i.e., its row sum $L(i) = 0$). If a random surfer lands on such a page, they cannot follow any links. To prevent them from getting "stuck" and to ensure PageRank distributes correctly, dangling nodes are handled specifically.
  - If $L(i) = 0$, I treat node i as if it links to every other node with equal probability $1/N$. This ensures that PageRank does not get trapped.

- **c. Create Row-Stochastic Matrix M:** I first construct an intermediate matrix $\mathbf{M}$ where M_{ij} is the probability of moving *from* node i *to* node j. This means the *rows* of $\mathbf{M}$ must sum to 1.
  - For a non-dangling node i:

$$M_{ij} = \frac{W_{ij}}{L(i)} = \frac{W_{ij}}{\sum_{k=0}^{N-1} W_{ik}}$$

  - For a dangling node i ($L(i) = 0$):

$$M_{ij} = \frac{1}{N} \quad \text{for all } j \in \{0, \ldots, N-1\}$$

The matrix $\mathbf{M}$ is:

$$\mathbf{M} = \begin{pmatrix} M_{00} & M_{01} & \cdots & M_{0,N-1} \\ M_{10} & M_{11} & \cdots & M_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ M_{N-1,0} & M_{N-1,1} & \cdots & M_{N-1,N-1} \end{pmatrix}$$

Each row of $\mathbf{M}$ sums to 1: $\sum_j M_{ij} = 1$. Thus, $\mathbf{M}$ is a **row-stochastic matrix**.

- **d. Transpose to Get Column-Stochastic Matrix P:** The PageRank power iteration formula relies on a matrix $\mathbf{P}$ where P_{ji} represents the probability of moving from node i to node j, and its *columns* sum to 1. This is achieved by transposing $\mathbf{M}$.

$$P_{ji} = M_{ij}$$

  So, $\mathbf{P} = \mathbf{M}^T$.

$$\mathbf{P} = \begin{pmatrix} M_{00} & M_{10} & \cdots & M_{N-1,0} \\ M_{01} & M_{11} & \cdots & M_{N-1,1} \\ \vdots & \vdots & \ddots & \vdots \\ M_{0,N-1} & M_{1,N-1} & \cdots & M_{N-1,N-1} \end{pmatrix}$$

  Each column of $\mathbf{P}$ sums to 1: $\sum_j P_{ji} = 1$. Thus, $\mathbf{P}$ is a **column-stochastic matrix**, ready for the PageRank formula.

- **Conceptual Visualization of P:** In the matrix $\mathbf{P}$, the columns represent the *source* nodes, and the rows represent the *destination* nodes. An element P_{ji} is the probability of a surfer moving from node i to node j.

$$\mathbf{P} = \begin{pmatrix} P_{00} & P_{01} & \cdots & P_{0,N-1} \\ P_{10} & P_{11} & \cdots & P_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ P_{N-1,0} & P_{N-1,1} & \cdots & P_{N-1,N-1} \end{pmatrix}$$

  where P_{ji} (element in row j, column i) is the probability of moving from node i to node j. The sum of probabilities in each column must be 1. For example, the sum of probabilities of moving from node i to any other node j is $\sum_j P_{ji} = 1$.

4. Damping Factor (α) and Teleportation Vector (v)

These components are crucial for ensuring the algorithm's convergence and reflecting the behavior of a random surfer.

- **Damping Factor (α):**
  - **Definition:** Represents the probability that the random surfer will continue to follow a link on the current page. From experience of previous research set to be 0.85.
- **Teleportation Vector (v):**
  - **Definition:** Represents the probability distribution over all pages that the random surfer jumps to when they get "bored" (i.e., decide not to follow a link, which occurs with probability $1 - \alpha$). In the standard PageRank model, this is a uniform distribution across all pages.
  - **Mathematical Representation:**

$$\mathbf{v} = \begin{pmatrix} 1/N \\ 1/N \\ \vdots \\ 1/N \end{pmatrix}$$

    This ensures that the elements of $\mathbf{v}$ sum to 1: $\sum_{i=0}^{N-1} v_i = 1$. If $(1 - \alpha)$ "triggers", the random surfer jumps to the edge with index chosen in v.

5. Power Iteration Initialization

The PageRank algorithm uses an iterative approach (power iteration) to find the steady-state PageRank scores. I start with an initial guess $r^{(0)}$.

- **Initial PageRank Vector ($\mathbf{r}^{(0)}$):**

- **Definition:** Each page is initially assigned an equal PageRank score.
- **Mathematical Representation:**

$$\mathbf{r}^{(0)} = \begin{pmatrix} 1/N \\ 1/N \\ \vdots \\ 1/N \end{pmatrix}$$

The sum of elements in $\mathbf{r}^{(0)}$ is 1: $\sum_{i=0}^{N-1} r_i^{(0)} = 1$ and N is the number of countries in the network.

- **Convergence Threshold (ϵ):**
  - **Definition:** A small positive value used to determine when the algorithm has converged. The iteration stops when the change in the PageRank vector between successive steps falls below this threshold.
  - **Typically:** $\epsilon = 10^{-8}$.

6. Power Iteration Loop

This is the core of the PageRank algorithm, where PageRank scores are iteratively refined until they stabilize.

- **Loop Condition:** The loop continues as long as the measured change (Δ) in the PageRank vector is greater than the convergence threshold ϵ.
- **Scoring Vector:** $\mathbf{r}^{(k)}$ represents the PageRank scores of all pages after the k-th iteration of the algorithm. More precisely, $\mathbf{r}^{(k)}$ can be interpreted as **the probability distribution** of the random surfer being on any given page after k steps of their journey across the web (or our network of countries). The vector $\mathbf{r}^{(k)}$ is an $N \times 1$ column vector:

$$\mathbf{r}^{(k)} = \begin{pmatrix} r_0^{(k)} \\ r_1^{(k)} \\ \vdots \\ r_{N-1}^{(k)} \end{pmatrix}$$

  where $r_i^{(k)}$ is the PageRank score (or the probability of the random surfer being on page i) after the k-th iteration. Crucially, because it's a probability distribution, the sum of all elements in $\mathbf{r}^{(k)}$ must be 1: $\sum_{i=0}^{N-1} r_i^{(k)} = 1$.

- **PageRank Update Equation:** For iteration $k+1$, the new PageRank vector $\mathbf{r}^{(k+1)}$ is calculated from the previous vector $\mathbf{r}^{(k)}$ using the formula:

$$\mathbf{r}^{(k+1)} = \alpha \mathbf{P} \mathbf{r}^{(k)} + (1-\alpha)\mathbf{v}$$

  Let's interpret the terms:

  - $\mathbf{P}$ is our column-stochastic transition matrix, where P_{ji} is the probability of a random surfer moving from page i to page j.$\mathbf{Pr}^{(k)}$**:** When you multiply the matrix $\mathbf{P}$ by the vector $\mathbf{r}^{(k)}$, you are essentially calculating where the surfer is likely to be *after one more step*, assuming they always follow a link according to the probabilities defined by $\mathbf{P}$. The j-th component of this product, $(\mathbf{Pr}^{(k)})_j = \sum_{i=0}^{N-1} P_{ji} r_i^{(k)}$, represents the total probability flow into page j from all other pages i, weighted by their current probabilities of being visited. In simpler terms, it's the sum of the votes page j receives, where each vote's strength is proportional to the PageRank of the voting page ($r_i^{(k)}$) and the probability of that page linking to j (P_{ji}).
  - $\alpha(\mathbf{Pr}^{(k)})$: This scales the PageRank obtained by following links by the damping factor α.

– $(1-\alpha)\mathbf{v}$: This term represents the PageRank added by the "teleportation" mechanism. A small portion of the total PageRank $(1-\alpha)$ is uniformly distributed among all pages via the vector $\mathbf{v}$. This is crucial for overcoming issues like "rank sinks" (groups of pages that only link internally) and ensuring that all pages (even those with no incoming links) can eventually gain some PageRank. It ensures that the underlying Markov chain is irreducible and aperiodic, guaranteeing a unique stationary distribution (the PageRank vector).

- **Convergence Check (L1 Norm):** The L1 norm (Manhattan distance) is used to quantify the overall change between the new and old PageRank vectors:

$$\Delta = \|\mathbf{r}^{(k+1)} - \mathbf{r}^{(k)}\|_1 = \sum_{i=0}^{N-1} |r_i^{(k+1)} - r_i^{(k)}|$$

  This Δ value indicates the stability of the PageRank distribution.

- **Update and Increment:** The newly computed PageRank vector $\mathbf{r}^{(k+1)}$ becomes the current vector $\mathbf{r}^{(k)}$ for the next iteration. The iteration counter is incremented.

7. Final Results

Once the power iteration loop converges ($\Delta \leq \epsilon$), the final $\mathbf{r}$ vector contains the stable PageRank scores for each country. These scores represent the calculated importance or authority of each country within the network, based on the link structure and weights.

12.2 Visualization of Trade

The function *visualize_trade* is defined as follows:

```
def visualize_trade(TN_t_c, pagerank_t_c, top_n=10, period_val):
```

It accepts the following parameters:

- `TN_t_c`: A Pandas DataFrame containing the edge data for a specific period. It is expected to have columns like `'reporterISO'`, `'partnerISO'`, and `'W_ij'`.
- `pagerank_t_c`: A Pandas DataFrame containing the pre-calculated PageRank scores for countries in the given period. It should have columns `'countryISO'` and `'PageRank'`.
- `top_n` (default: 10): An integer specifying the number of top-ranked countries (by PageRank) to be placed in the central cluster.
- `period_val`: An integer representing the specific period being visualized. Used primarily for the plot title.

Graph Initialization and Edge Addition

The function begins by creating a directed graph using `networkx.DiGraph()`. This is suitable for representing trade flows from reporter to partner.

```
    G = nx.DiGraph()
    for index, row in df_period.iterrows():
        reporter = row['reporterISO']
        partner = row['partnerISO']
        weight = row['W_ij']
        G.add_edge(reporter, partner, weight=weight)
```

For each row in the input `TN_t_c` DataFrame, an edge is added to the graph `G`. The `W_ij` value is stored as a `'weight'` attribute for each edge, which will later be used to determine the link width. A check is performed to ensure the graph is not empty before proceeding with visualization:

```python
    if not G.nodes():
        print(f"No network data found for period {period_val}. Cannot visualize.")
        return
```

PageRank Data Preparation

The function then prepares the PageRank data for use in layout and visualization:

```python
    active_nodes = list(G.nodes())
    pagerank_t_c_active = pagerank_t_c[pagerank_t_c['Country'].isin(active_nodes)
        ].copy()
```

This step ensures that only PageRank scores for countries actually present in the current network (‘G.nodes()‘) are considered, preventing issues if ‘pagerank_t_c‘ contains countries not relevant to the current ‘TN_t_c‘ slice. This should not occur since I calculate the pagerank specifically for this period t and capability c every time.

Identifying Top and Other Countries

Countries are categorized into two groups based on their PageRank:

```python
    if len(pagerank_t_c_active) > top_n:
        top_countries_df = pagerank_t_c_active.head(top_n)
        other_countries_df = pagerank_t_c_active.iloc[top_n:].copy()
        other_countries_df = other_countries_df.sort_values(by='PageRank',
            ascending=False)
    else:
        top_countries_df = pagerank_t_c_active
        other_countries_df = pd.DataFrame(columns=['Country', 'PageRank'])
```

If the total number of active countries is greater than `top_n`, the top `top_n` countries (by PageRank) are assigned to `top_countries_df`, and the rest to `other_countries_df`. The `other_countries_df` is explicitly sorted by PageRank in descending order, which is crucial for their arrangement on the outer circle. If there are fewer countries than `top_n`, all are considered "top" and will be centrally placed.

Custom Positions

This is the core of the custom layout. A dictionary `pos` is created to store the (x, y) coordinates for each node.

```python
    pos = {}
    center_radius = 1.0  # Radius for the central cluster
    outer_radius = 2.0   # Radius for the outer circle
    num_top = len(top_countries_df)
    num_others = len(other_countries_df)
```

`center_radius` defines how tightly packed the central nodes are, and `outer_radius` defines the size of the larger circle.

```python
    for i, country in enumerate(top_countries_df['Country']):
        angle = 2 * math.pi * i / num_top if num_top > 0 else 0
        x = center_radius * math.cos(angle)
        y = center_radius * math.sin(angle)
        pos[country] = np.array([x, y])
```

The `top_n` countries are arranged in a small circle around the origin $(0, 0)$. Their angular positions are evenly distributed, and their radial distance is determined by `center_radius`.

```python
for i, country in enumerate(other_countries_df['Country']):
    angle = 2 * math.pi * i / num_others if num_others > 0 else 0
    x = outer_radius * math.cos(angle)
    y = outer_radius * math.sin(angle)
    pos[country] = np.array([x, y])
```

The remaining countries are positioned on a larger circle of radius `outer_radius`. Their positions are also evenly distributed around this circle, ordered according to their PageRank (due to the prior sorting of `other_countries_df`).

Visualization Parameters and Drawing

`matplotlib.pyplot.figure()` is used to create the plot area.

12.2.1 Node Sizing and Coloring

```python
node_pageranks = pagerank_t_c_active.set_index('Country')['PageRank']
max_pr = node_pageranks.max()
min_pr = node_pageranks.min()
node_sizes = [
    1 + 3000 * ((node_pageranks[node] - min_pr) / (max_pr - min_pr + 1e-9))
    for node in G.nodes()
] if len(node_pageranks) > 1 else [2000 for _ in G.nodes()]
node_colors = ['red' if node in top_countries_df['Country'].values else '
    skyblue' for node in G.nodes()]
```

- `node_sizes`: The size of each node is made proportional to its PageRank score. A base size of 1 is added, and then scaled up to a maximum of 3001. This visually emphasizes countries with higher centrality.
- `node_colors`: Top countries are colored `red` to visually distinguish them from other countries, which are colored `skyblue`.

Edge Widths

```python
edge_weights = [G[u][v]['weight'] for u, v in G.edges()]
max_weight = max(edge_weights) if edge_weights else 1
min_weight = min(edge_weights) if edge_weights else 0
edge_widths = [
    0.01 + 30 * ((w - min_weight) / (max_weight - min_weight + 1e-9))
    for w in edge_weights
] if len(edge_weights) > 1 else [2.0 for _ in G.edges()]
```

The width of each edge (link) is made proportional to its `W_ij` (trade volume) weight. A base width of 0.01 is added, and then scaled up to a maximum of 30.01. This allows for visual identification of stronger trade connections.

Drawing Elements

- `nx.draw_networkx_nodes()`: Draws the nodes using the calculated `pos` dictionary for positions, and applies the custom sizes and colors.
- `nx.draw_networkx_edges()`: Draws the directed edges. It uses the custom calculated `edge_widths`.
- `arrowsize`: Controls the size of the arrowheads.

- `min_source_margin` and `min_target_margin`: These parameters offset the start and end points of the arrows from the exact center of the nodes, making the arrowheads more visible, especially for larger nodes.
- `connectionstyle="arc3,rad=0.1"`: This gives edges a slight curvature.
- `nx.draw_networkx_labels()`: Adds the country ISO codes as labels to the nodes.

```
    plt.title(f'Trade Network for Period {period_val}\\nTop {top_n} Countries 
        Centered by PageRank', size=16)
    plt.axis('off')
    plt.axis('off')
    plt.show()
```