Master Thesis in Engineering

Study of Innovation Trend in Software Service Networks

- Trend Analysis based on Social Network Analysis -

소프트웨어 서비스 네트워크의 혁신 경향성 연구
: 사회 연결망 분석에 의거하여

February 2014

Graduate School of Seoul National University

Technology Management, Economics, and Policy Program

Wool rim, Lee

Study of Innovation Trend in Software Service Networks

- Trend Analysis based on Social Network Analysis -

지도교수 **Jörn Altmann**

이 논문을 공학석사학위 논문으로 제출함

2014 년 2 월

서울대학교 대학원

협동과정 기술경영경제정책 전공

이울림

이울림의 공학석사학위 논문을 인준함

2014 년 2 월

위 원 장 **Junseok Hwang** (인)

부위원장 **Jörn Altmann** (인)

위 원 **Kibae Kim** (인)

Abstract

Study of Innovation Trend in Software Service Networks

Service networks can be considered open innovation systems. As software vendors provide their software as a service (SaaS) and allow users to access its functions through open interfaces, innovation style has shifted from local innovation to collective innovation. This new innovation trend has led to the research on the structural and the evolutionary patterns in SaaS networks. However, prior research that mostly focuses on the static properties of network structure and position of nodes misses the dynamics of evolving network. In this thesis, the changes of network positions over time are studied through the investigation of the trend of the centralities of representative software services in SaaS networks. The results suggest that each software service follows a typical life cycle in its network position, and that the innovation trend shifts from the photo to social networking services by involving a transition of network structure. These results imply that innovation studies should consider not only static network topology and position, but also the changing pattern of position in evolving network.

Keywords: Service Network, Software-as-a-Service, Innovation Trend, Centralities, Social Network Analysis, Composite Services, Open Innovation.

Student Number: 2012-21042

Contents

List of Tables

List of Figures

Chapter 1. Introduction

1.1 Motivation

The advancement of IT technology enabled the provision of software services on the Internet. The demand for new business models that motivate users to innovate is underpinned by the forces that allow developers to upgrade and combine services at no charge. In this context, the network of combined software services represents an open innovation environment (Chesbrough, 2003). Examples of open innovation systems include databases of academic journals, which allow researchers to contribute their research articles, as well as read and reference the research papers by others (Wagner and Leydesdorff, 2005); the community for open source developers, which stimulates the exchange of information on software development projects (Valverde and Sole, 2007); and the network of software services with open APIs, which allows accessing service data and functions (Kim et al., 2011).

Software-as-a-Service (SaaS) is an old paradigm. Even though a computer can be distanced from other computers, users can utilize software services through the Internet. Since the emergence of cloud computing, SaaS is spotlighted again ranging from office to computer resources (Campbell-Kelly, 2009). SaaS can be served as a final good or a resource to users, and users utilize its original function and data for their innovation (Haines and Rothenberger, 2010). As software vendors provide their software as a service and allow users to access the functions via open interfaces, users can reuse the existing services for creating new services.

Because open innovation on service networks promotes the reuse of existing innovation resources, the innovation studies are interested in the structure of the entire innovation system (Chesbrough, 2011; Maglio, 2006). One of the main efforts of these studies is to apply network analysis to the network of innovation agents and resources. It follows the research on social networks (Freeman, 1979), statistical physics (Albert et al., 1999), and co-authorship networks (Newman, 2001; Wagner and Leydesdorff, 2005). The new trend of innovation, in which each service is reused by other services, invites researchers on information systems to analyze the structure and the evolution of these information systems as a platform for “open innovation” (Chesbrough, 2003).

1.2 Problem Description

An important part in prior innovation research on networks has been to investigate the structure and the evolutionary pattern of networks (Newman, 2001; Valverde and Sole, 2007; Wagner and Leydesdorff, 2005) and to correlate the network characteristics with innovation performance (Granovetter, 1973; Grewal et al., 2006; Krackhardt and Stern, 1980). The innovation research examined the relationship between a node’s social network position in a network structure and its innovation performance (Granovetter, 1973; Krackhardt and Stern, 1980). Similar analyses have been performed for innovation communities (Grewal et al., 2006; Valverde and Sole, 2007) and innovation resources (Kim et al., 2011).

However, previous studies have considered the network properties as static, and their effect on innovation as invariant. In this sense, these studies miss the complex and dynamic behaviour of each node in an evolving network. This thesis complements prior research in that it explores the dynamic behaviour of an innovation system by analyzing the position trends of representative software services in the evolving SaaS network depending on time.

1.3 Research Goal

The main objectives of this thesis are to introduce the dynamic behavior into the innovation studies, to investigate the trend of centralities of representative software services in a SaaS network, and to discuss the patterns of changing positions of software services in the evolving SaaS network.

1.4 Research Questions

Two main questions are adduced to achieve the research goal: (1) Is the position of a node fixed in a network? (2) What does a position change imply for innovation? To answer these two main questions, the following four sub-questions (two per each of the main questions) need to be addressed, respectively: (1a) Which software services locate at the central position? (1b) Do software services at the central position shift? (2a) Which service is more appropriate for innovation in the evolving network? (2b) Is the innovation trend of a service related to service category?

1.5 Research Outline

The remainder of this thesis is organized as follows. The next chapter gives a brief description of the conceptual background of SaaS innovation systems and the network perspectives of innovation studies.

Chapter 3 describes the collected data and how to form the SaaS network with the data, and explains how to choose representative software services and measure the centralities.

The software service network considered in this study has been built from the empirical data surveyed on the public Web site www.programmableweb.com. This network consists of nodes and links. Each node represents a software service, and each link denotes the joint use of two existing software services for the development of new software services. Software services with open application programming interfaces (APIs) can be used for creating new composite software services. The representative software services have been selected from 7427 software services according to the frequency of their use throughout the study period.

The normalized degree centrality, the normalized betweenness centrality, and the normalized eigenvector centrality of the representative software services have been measured to examine their position over time. These results allow examining whether the innovation leader shifts from one service to another. The measurement of the normalized degree centrality and the normalized betweenness centrality for the four most frequently used software services in the social networking service category has been added. It can help reveal whether the shift of an innovation leader’s position is related to the innovation of a group of services (i.e., a category).

Chapter 4 presents the results of the analysis, specifically, the trend analysis of nodes' positions of the representative software services within the SaaS network. The main results of the research are as follows. First, similarly to the innovation adoption life cycle (Rogers, 2003), each software service exhibits a life cycle including the growing, maturity, and decline phase. Second, some software services achieve the hub positions in the entire network, and some software services achieve the core positions within a cluster of software services. It can be noted that the photo service (i.e., Flickr) has been replaced by a social networking service (i.e., Twitter), while the mapping service (i.e., Google Maps) maintains its hub position in the network. Finally, the innovation adoption life cycle of a software service depends on the innovation trend of its service category. The reason for this can be found in the fact that some of the software services in the same category act as complements to the most frequently used services, pushing the success of those most frequently used services even further.

The findings of this study have entrepreneurial implications with regard to the positioning strategy of software service vendors. Furthermore, these are also expected to redirect the network analysis of an innovation system from a static perspective to a dynamic perspective. The conclusions and implications of this study are explained in Chapter 5.

Chapter 2. Literature Review

2.1 Open Innovation in Software Service Network

As IT technology advanced and new business models emerged that motivate consumer to participate in an innovation process (e.g., Web 2.0 service creation), software vendors offered their software as a service (SaaS). SaaS is a paradigm, which allows users to run software that is installed remotely via the Internet. The SaaS paradigm emerged with commercial computing in the 1980s but has been in the downturn with the growth of personal computer. Since the emergence of cloud computing, SaaS is spotlighted again in a various areas ranging from office software (e.g., Google docs) to computing resources (e.g., Amazon S3) (Campbell-Kelly, 2009).

One type of SaaS implementation is based on the Web services under the concept of Service Oriented Architecture (SOA). SOA defines how users can compose and reuse services through the open interfaces. It is called open Application Programming Interface (API), and one of the famous example is Google Maps (Haines and Rothenberger, 2010; Papazoglou and Georgakopoulos, 2003). A composite service is created by adding a unique value to an existing software service or by integrating several existing software services. This composite service is called “mashup” (Ogrinz, 2009).

Figure 1 is an example of a composite service, Weather Bonk. It provides weather information on a map by accessing open APIs of Google Maps and Weather Bug. According to the definition of software service network, Weather Bonk service connects all eight software services with each other (Figure 1), resulting in a fully connected graph.

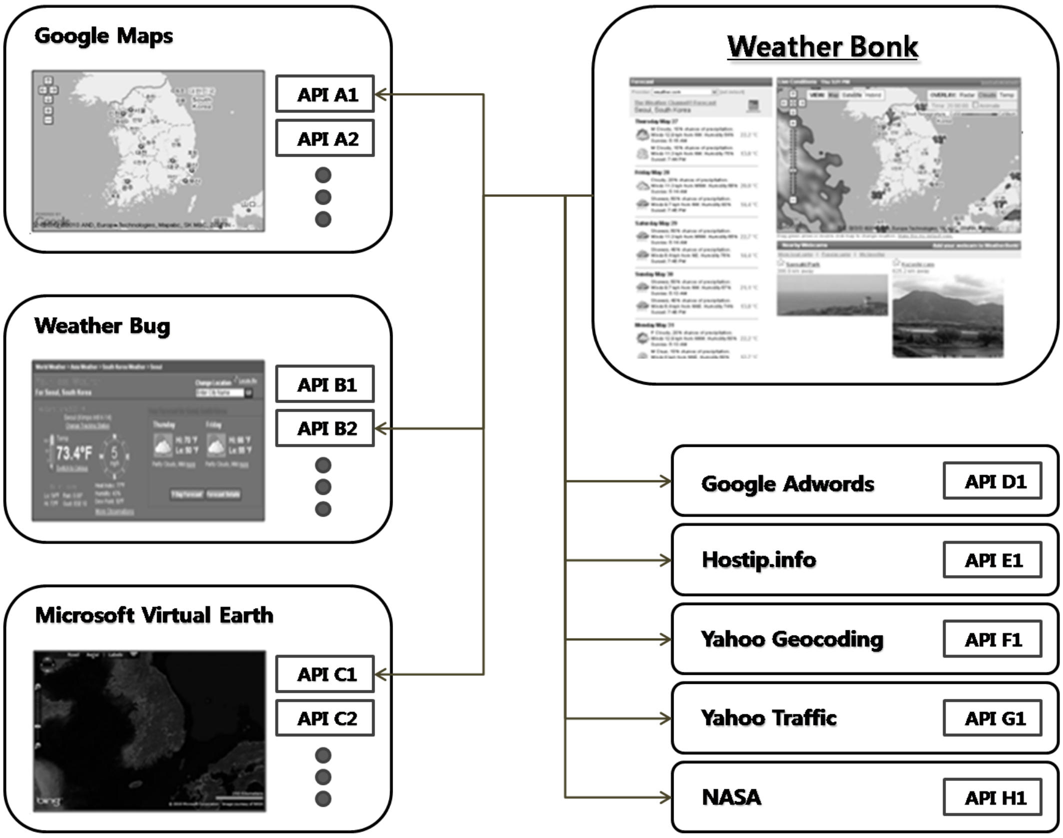

Figure 1. Example of a composite service, Weather Bonk

In this architecture, software vendors achieve a huge scale of innovation utilizing their service users (O'Reilly, 2007). This is open innovation in a software service environment. Innovation occurs through the free sharing of software services with their service users and even with their competitors (Chesbrough, 2003). Any software service provider can participate in the innovation by simply opening up the APIs. This means that users can reuse the data and functions of those services for their own service development, any users can join in this kind of innovation. This is the reason why the innovation system grows with a specific structure.

2.2 Network Position and Innovation

In the social network studies, a diversity of measures has been developed to analyze the position of nodes (agents) in a network. The position of an agent in a network is important, because the position has a relationship with the role of the agent in the society (Scott, 1991). The most popular measure is centrality based on the degree of a node or the geodesic between two nodes (Freeman, 1979). The degree centrality of node is measured by the number of adjacent nodes, which implies how deeply a node is embedded among its neighbours. The betweenness centrality of a node considers the number of shortest paths (geodesics) through the node compared to all possible geodesics. It is used to determine the extent to which the node is interconnected with other nodes. If a node has high betweenness centrality and low degree centrality, it can be supposed that the node might bridge several separated communities within the network (Everard and Henry, 2002). In addition, it is also important to understand which nodes are connected. As a node's power in a network is determined by the power of its neighbors (Bonacich, 1987), the eigenvector centrality has been defined. The eigenvector centrality indicates the sum of the degree centralities of the adjacent nodes.

Empirical research about the network reveals that the degree distribution of node in some real networks shows the scale-free property. The scale-free property represents that the frequency of node degree decays by the power function. This distribution is very inhomogeneous, or highly skewed, compared with the exponential distribution of random networks or the distribution of regular networks (Albert et al, 1999). In a scale-free network, few nodes are connected with a majority of nodes in order to make the distance between nodes small, while the degree of most nodes is low. The few nodes with high node degree are called "hubs".

Since innovation performance of an agent is closely related to its network position, the heterogeneity of network position is important for the study of innovation in the network (Burt, 1992; Granovetter, 1973; Grewal et al., 2006). As the network position of an agent can be quite different, the performance of an agent can vary considerably. Innovation is a recombining process of fragmented existing knowledge (Hargadon 2002). An agent (e.g., a researcher or a firm) can create new knowledge by recombining knowledge of the group. If an agent bridges two distant and separate clusters within the network through (even) low and infrequent connections with agents which belong to these clusters, the agent enables an innovation. The agent can access diverse knowledge efficiently and effectively so that new ideas can emerge from the whole system, not a clustered group only (Burt, 1992; Granovetter, 1973).

Prior research on the analysis of innovation in the network overlooks the dynamic aspect of network evolution. It just concentrates on static property and statistical relationship between nodes. For instance, in the empirical network analysis and the evolutionary model introduced by Albert et al. (1999), the characteristics of scale-free networks are invariant and the network is formed by a preferential attachment rule that determines the link between a new node and existing nodes. According to this model, a node is located at the central position forever, if it has been selected as a hub at the early stage.

Innovation research exploring the effect of network position on innovation also assumes that the network structure does not change in the evolutionary network. Granovetter (1973) and Burt (1992) emphasized the importance of nodes which are connected with separate and distant clusters for innovation under the fixed network structure. The network position affecting innovation has been investigated in various conditions specifying the relationship between network position and innovation performance. For example, the more central a node (agent) in a network is, the higher the innovation performance of the node (agent) is under the conditions that the node has a good absorptive capacity (Tsai, 2001), that the node is in a central group (Sasidharan et al., 2011), and that the node has weak ties (Hansen, 1999).

In these literatures, the analyses were carried out with the data which represent snapshot of network structure and innovation performance. These analyses of Granovetter (1973), Burt (1992), Tsai (2001), Sasidharan et al. (2011), and Hansen (1999) did not consider the time factor in their discussion concerning the relationship between network position and innovation.

Innovation systems are more dynamic than the network studies mentioned. On the one hand, innovation systems of diverse levels (e.g., a single technology or entire industry) show life cycle behaviour from emergence to decline through maturity. These are described by the models that explain the driving force of the rise and fall of an innovation system through the diffusion of technology (Bass, 1969; Rogers, 2003), the opportunity caused by a new technology and the competition (Jovanovic and MacDonald, 1994), and the opportunity of strategic alliances with competitors (Lemmens, 2004). On the other hand, especially in case of innovation through collective intelligence, the trend of innovation varies as the interest of the crowd shifts (Jin et al., 2009). The changing interest of the crowd is often revealed in the service networks including Web sites, blogs, and some special forums, and the analysis of the changing interest can be used to predict the innovation trend (Gloor et al., 2009).

As a result of these perspectives, the question rises as to whether the structure of an innovation network remains stable while the internal status of a node in the network changes. One of the research consequences about the question shows that the hubs in the software service network are replaced while the scale-free topology is kept (Hwang et al., 2009).

Chapter 3.   Methodology

3.1 Service Network

The data has been gathered from the Web site www.programmableweb.com which lists information about software services. The information includes the name, the launch date, and the brief description of the services, the openness of the service's APIs, and the services that reuse the existing services. If a software service is composed, this software service is called mashup in Web site. The information of all software services has been collected since the first composite service was added on September 14th, 2005. The last data set for the research is September 30th, 2012.

The software service network is defined as a set of nodes and links between these nodes. Each node represents a software service that opens up its APIs, and a link indicates the existence of a composite service that uses the connected nodes. The creation of a composite service yields a complete graph of software services that is used for the development of the composite service. For instance, shown in Figure 1, the creation of Weather Bonk generates 28 links between the eight software services with open APIs (Google Maps, Microsoft Virtual Earth, Weather Bug, Google Adwords, hostip.info, Yahoo Geocoding, Yahoo Traffic, and NASA), connecting all eight nodes with each other.

The links of the software service network are non-directional and weighted. That is, a link does not show the information of the source and destination of its relationship, but displays the frequency of use of the link. To illustrate, assuming that a new composite service is created using Google Maps and Yahoo Traffic in addition to the SaaS network including the Weather Bonk, the weight of the link between Google Maps and Yahoo Traffic is 2 while the weight of other links is still 1.

3.2 Measures

While the software service network is a weighted graph, the centrality measures have usually been defined for binary graphs (Bonacich, 1987; Everard and Henry, 2002; Freeman, 1979). Therefore, it is necessary to modify the centralities for weighted graphs (Opsahl et al., 2010). Following terminologies are introduced in this thesis. Let w_{ij} be the weight of the link between nodes i and j belonging to the network G whose size is g. The weight of a link in a SaaS network denotes the occurrence frequency of collaboration between the two nodes, and w_{ij} is an integer in a SaaS network. If w_{ij} is zero, there is no link between the two nodes. Therefore, the weight represents the frequency of interaction between the nodes.

In a weighted graph, the degree centrality is defined as the sum of the weights of the links that a node has. That is, the degree centrality of node i in a weighted graph is the sum of weights of the links w_{ij} that node i has with node j. The degree centrality of a node in a weighted graph is also called the strength of the node because it illustrates the strength of connectivity of the node (Opsahl et al., 2010). The degree centrality seems to be increased as the network size increases even in the networks with identical density. In order to eliminate the effect of network size from degree centrality, the degree centrality of binary graph is normalized by the maximum possible number of links that a node can have. The normalized degree centrality in a binary graph varies between *0* and *1*. If each node is linked with all the other nodes, the degree centrality goes to *1*. If a node is not connected with any other nodes, the degree centrality goes to *0*. The degree centrality of weighted graph is also normalized by the maximum possible number of neighbours that a node can have (i.e., g-1):

$$s'(i) = \mathrm{CD}'(i) = \sum_{j \neq i} w_{ij} / (g-1). \qquad \text{Eq. (1)}$$

However, applying the normalized degree centrality to a weighted graph has a limitation. If the weight is larger than *1*, the normalized degree centrality can become larger than *1*. In spite of that, the normalized degree centrality is a good measure to compare the centralities of nodes in the different networks of various sizes.

The betweenness centrality is defined on the basis of the shortest paths (geodesics). In this thesis, the betweenness centrality for a weighted graph is employed instead of the betweenness centrality for a binary graph in common with the degree centrality. Thus, the shortest path length in a weighted graph should be re-defined. In general, the shortest path between two nodes is the path passing the smallest number of links between them, and the shortest path length is the number of links of the shortest path. In a weighted graph, a pair of nodes is said to be closer as the weight of link between nodes gets larger in the software service network. For this reason, it is reasonable to assume that between any two nodes the path length considered the weight is smaller than the path length without considering. Therefore, the shortest path length $d^w(i,j)$ between two nodes i and j in a weighted graph G is determined as the sum of the inversed weights of the links on the shortest path between the nodes (Opsahl et al., 2010):

$$d^w(i,j) = \min \{|p_{ij}| \textit{ for all } p_{ij} \textit{ where } |p_{ij}| = 1/w_{ih} + \cdots + 1/w_{kj}\}. \qquad \text{Eq. (2)}$$

Let σ_{ij}^w be the number of shortest paths between node i and j in a weighted graph G, and $\sigma_{ij}^w(v)$ be the number of shortest paths between node i and j passing through the node v. The betweenness centrality of node v in a weighted graph is defined as the sum of $\sigma_{ij}^w(v)$ / σ_{ij}^w for all nodes i and j in the same manner as the betweenness centrality in a binary graph (Freeman, 1979). Since the number of pairs of nodes increases as the network size increases, the betweenness centrality needs to be normalized by the maximum possible number of pairs of any two nodes except for the node v in a weighted graph G with size g, $(g-1)\cdot(g-2)/2$. Therefore, the normalized betweenness centrality for a weighted graph is defined as:

$$\mathrm{CB}'(i) = \sum_{i,j\neq i} \sigma_{ij}^w(v) / \sigma_{ij}^w / ((g-1)\cdot(g-2) / 2). \qquad \text{Eq. (3)}$$

The normalized betweenness centralities for a binary graph and a weighted graph vary between 0 and 1. If the shortest path passing through the node does not exist, the normalized betweenness centrality of the node is *0*. If all the shortest paths pass through the node, the normalized betweenness centrality of the node is *1*.

Lastly, the definition of eigenvector centrality is needed. The eigenvector centrality of a node is the sum of the eigenvector centralities of its adjacent nodes. The adjacency of each node in a binary graph whose network size is g is represented as a g by g matrix $\boldsymbol{A}$. If node i and node j are directly connected, the element of matrix $\boldsymbol{A}$, a_{ij}, is *1*, and if not, the element of matrix $\boldsymbol{A}$ is *0*.

In this thesis, $\boldsymbol{x}$ is the vector of the eigenvector centralities of all nodes in the graph, which should satisfy the condition $\boldsymbol{A}\ \boldsymbol{x} = \lambda\ \boldsymbol{x}$. The symbol λ indicates an eigenvalue (Bonacich, 1987). The eigenvector centrality is normalized by the vector length $||\boldsymbol{x}||$, and the eigenvector centrality of a node in a weighted graph is the sum of the eigenvector centralities of all its adjacent nodes. Let $\boldsymbol{W}$ be a g by g matrix where element w_{ij} is the weight between nodes i and node j in a graph G with size g. The normalized eigenvector centrality of node i is defined as the i-th element of $\boldsymbol{x}$, satisfying the following condition:

$$\mathbf{W}\ \boldsymbol{x} = \lambda\ \boldsymbol{x}. \qquad \text{Eq. (4)}$$

The number of eigenvalues is g at most. The eigenvector with the largest eigenvalue among the multiple eigenvalues is selected because it involves the nodes in the main component that consists of the largest amount of nodes segregated from the other components. And the eigenvector centralities of all nodes which are not belong to the largest component are zero.

Chapter 4. Analysis Results

The data set has been surveyed from www.programmableweb.com which lists information about software services. The information includes the name, the launch date, and the brief description of the services, the accessibility of the APIs, and the services used for creating the composite services. The data have been collected from September 14th, 2005 when the first composite service was added to September 30th, 2012. In this period, 7427 software services have been registered, 6780 services are composite services among the registered software services. The composite services were developed by utilizing 1153 software services that offered their functionalities through the open APIs. From the surveyed data, a software service network for the analysis consists of 1153 nodes and 23573 links. This network is undirected and weighted, and the weight of the link between a pair of nodes represents the number of composed software services based on these two nodes.

For the analysis, eight representative software services are selected according to the frequency of use during the study period. Among them, four services (i.e., Google Maps, Twitter, YouTube, and Flickr) are the most frequently used services in the software service network, and the remaining four services (i.e., Facebook, Foursquare, LinkedIn, and Facebook Graph) and Twitter are the most frequently used services in the social networking service category. The social networking service category is chosen because it becomes popular later than the other service categories such as mapping, video, and photo.

The software services in the social networking service category are expected to show the different trends in their position compared with the four most frequently used software services. Before measuring the centralities, the brief introduction about selected eight representative software services is indicated in Table 1.

Table 1. Description of the selected software services

Service Name	Provider	Service Category	Entrance Time	Number of Uses
Google Maps	Google	Mapping	September 2005	2263
Twitter	Twitter	Social Networking	January 2007	644
YouTube	Google	Video	April 2006	598
Flickr	Yahoo	Photo	September 2005	590
Facebook	Facebook	Social Networking	August 2006	352
Foursquare	Foursquare	Social Networking	January 2010	82
LinkedIn	LinkedIn	Social Networking	February 2008	46
Facebook Graph	Facebook	Social Networking	December 2009	37

The research objective is to investigate the change of network position over time. The remainder of this section demonstrates the analysis results of the network position trends concerning the selected eight software services. The network positions of all eight services are determined by the normalized degree centrality and the normalized betweenness centrality. In addition, the eigenvector centralities of four services which are the most frequently used in the software service network are measured in order to complementing the analysis result. However, the eigenvector centrality of the social networking service category is ignored in this chapter because it does not show any notable trends.

The software service network in this research is unidirectional and weighted. The centralities are measured monthly, and the starting point of all services is September 2005. For example, the service network of May 2011 consists of all nodes and links from September 2005 to May 2011.

4.1 Network Position Trends of Most Frequently Used Services

4.1.1 Degree Centrality

The normalized degree centralities of four most frequently used software services (Google Maps, Twitter, Flickr, and YouTube) are measured respectively, in order to investigate how deeply each selected service is embedded among its neighbouring nodes in the software service network and how the embeddedness changes depending on time.

Figure 2 illustrates the trends of the normalized degree centralities during the study period and shows that the trends of four representative services shine out compared to the average normalized degree centrality of all other services (i.e., total) in the software service network. The normalized degree centralities of Google Maps, Flickr, and

YouTube show a similar tendency. Each of these services increases fast at the early period, and then decline until the end of the study period after staying at a certain level (i.e., prosperity period). The normalized degree centrality of Google Maps approximately increases from 0.65 in September 2005 to 2.70 in December 2010, but the growing rate is decreasing greatly since December 2007. After December 2010, the normalized degree centrality declines from 2.70 to 2.28 at the end of the study period. Likewise, the normalized degree centrality of Flickr soars from 0.10 in September 2005 to 2.01 in December 2007, and the normalized degree centrality of YouTube rises from 0.14 in January 2006 to 1.85 in September 2008. After the short period of slight increase, the normalized degree centralities of Flickr and YouTube decline from 2.18 in May 2010 to 1.68 and from 2.02 in January 2011 to 1.75 until the end of study period, respectively. Twitter is introduced later than the other three software services. The normalized degree centrality of Twitter remains stable at about 0.30 until January 2008, and grows fast to 1.75 in May 2011. After growing, it stays stable again with a slight decrease until the end of study period.

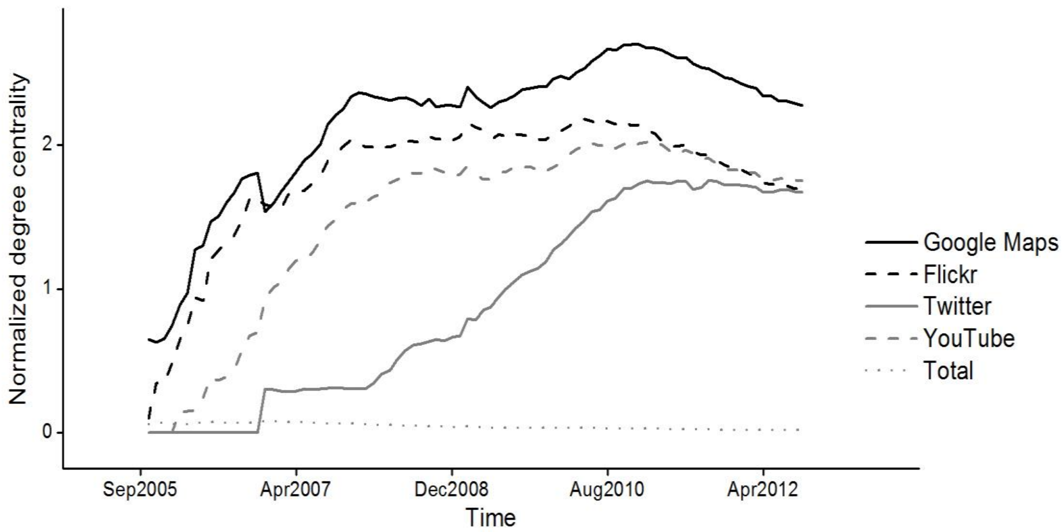

Figure 2. Trends of the normalized degree centralities of the four most frequently used software services

The trends of normalized degree centralities of the selected software services look like a life cycle of technology. Prior research on technology diffusion or user's adoption of technology suggests that a new technology is adopted exponentially after an inactive early period and then slows down, so that the cumulated adoption rate of the technology shows an S-like curve (Bass, 1969; Rogers, 2003). For the software service network, if a software service is frequently reused for developing the composite service, this service reaches the central position. But if a software service is not reused anymore or frequently, this service is pushed out to the periphery of the network. From this perspective, Google Maps, Flickr, and YouTube have grown to their maturity phase during the study period, and now face their saturation period. Twitter is in the maturity phase at the end of the study period because Twitter gets in to the software service network late.

In this research, Google Maps, Flickr, and YouTube do not show an early inactive period. This tendency might be related to the fact that these three services enter the network at the beginning of the evolution of service network. At that point, these services are already known to users, and users already know how to utilize these services. Therefore, these services do not need to wait for being diffused by "imitators" (Bass, 1969). However, Twitter displays an early inactive period after entering the service network. Users have to learn the value of Twitter as it is not the popular service at the early stage.

4.1.2 Betweenness Centrality

The normalized betweenness centralities are calculated to find out whether the four representative software services are bridges for other software services in the software service network, and to analyze the trends of the betweenness centralities of the selected services. The results of the normalized betweenness centralities are described in Figure 3. This shows that the trend of each software service is idiosyncratic. The normalized betweenness centrality of Google Maps is substantially higher than the other three services through the whole study period. It remains at about 0.22 on average after a slight drop. The normalized betweenness centrality of Flickr goes down slowly after the peak of the initial period. It rises rapidly from 0.00 in November 2005 to 0.18 in June 2006, and then decreases to 0.07 at the end of the study period with some fluctuations. The normalized betweenness centrality of YouTube increases gradually during the first half period, and keep the same level during the second half period. To be more specific, it grows from 0.00 in April 2006 to 0.09 in October 2009, and stays at this level after that. The normalized betweenness centrality of Twitter has another interesting trend. It remains 0.00 between December 2006 and January 2008, and jumps to 0.03 in July 2008. It jumps again from 0.03 in March 2009 to 0.08 in June 2009, and then gradually increases to 0.12 by the end of study period.

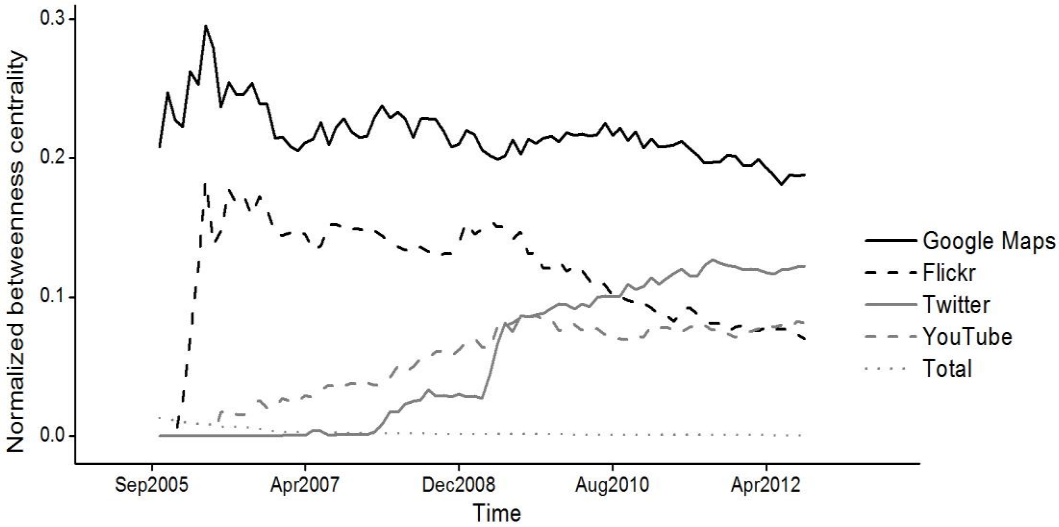

Figure 3. Trends of the normalized betweenness centralities of the four most frequently used software services

The shift of innovation trend can be demonstrated by the trend of normalized betweenness centrality. A node with high betweenness centrality and low degree centrality implies that the node bridges multiple separate and distant clusters, while a node with high betweenness centrality and high degree centralities occupies the hub position (Everard and Henry, 2002). In this software service network, Google Maps and Flickr are the hubs initially because they are connected to a lot of nodes and clusters in the network. Over time, normalized betweenness centrality of Flickr is declined while Google Maps keeps its level. Twitter increases in both normalized degree centrality and normalized betweenness centrality. From August 2010, Twitter even becomes a more important hub than Flickr. Flickr loses its initial hub position and becomes at the same level as YouTube in September 2012. Twitter, as a social networking service, could interconnect many more clusters of services than Flickr could. It is also notable that the normalized betweenness centrality of YouTube grows late while its normalized degree centrality is already high in the early periods. This means that YouTube is linked with many nodes but not connected with many clusters. In a graph theoretical perspective, it can be stated that YouTube is a core in some clusters, not a hub in the global network.

Considering the results of both normalized degree centrality and normalized betweenness centrality, Google Maps is expected to maintain its structural position for a considerable period of time. Innovation will happen through the use of Google Maps together with other major services (e.g., YouTube, Flickr, or Twitter). However, not all services can keep their position. As the normalized betweenness centrality of Flickr represents, it is likely that Flickr will not be able to keep the position of a major player in the near future. Flickr loses its attraction. The place left by Flickr will be substituted with

Twitter, because both normalized degree centrality and normalized betweenness centrality of Twitter increase gradually. It leads to the creation of new composite services. Therefore, the result is summarized that the innovation trend shifts from a photo service (i.e., Flickr) to a social networking service (i.e., Twitter).

4.1.3 Eigenvector Centrality

The eigenvalues and the eigenvectors of four representative software services are calculated, and the eigenvector which has the largest eigenvalue is select. The results are classified as three types of trends: shock, fast growth, and late growth. The normalized eigenvector centralities of Google Maps and Flickr start with high value (shock). Excluding the early fluctuation, the normalized eigenvector centralities of Google Maps and Flickr are about 0.44 and 0.42 on average during the study period. The normalized eigenvector centrality of YouTube grows fast during the initial period (fast growth). It increases from 0.08 in January 2006 to 0.42 in October 2008, and remains at 0.41 on average until the end of the study period. The normalized eigenvector centrality of Twitter goes up slowly compared to the other three services (late growth). It rises from 0.06 in December 2006 to 0.34 in September 2012.

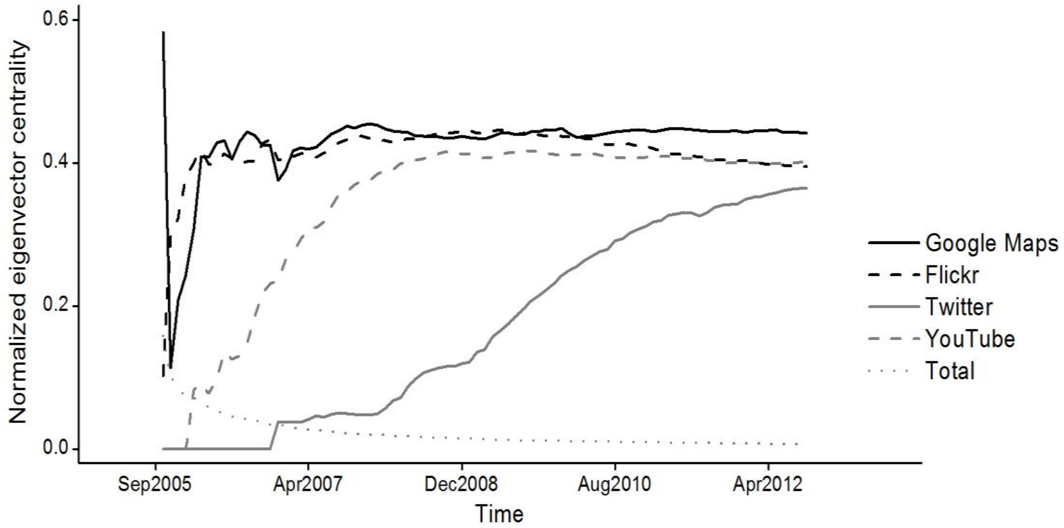

Figure 4. Trends of the normalized eigenvector centralities of the four most frequently used software services

The trends of normalized eigenvector centralities of the selected software services imply that the innovation in the software service network evolves in a complicate manner. As the normalized eigenvector centrality of a node indicates the strength of neighbouring nodes, it can be said that Google Maps, Flickr, and YouTube are stably linked with many and powerful nodes. Considering the result of normalized eigenvector centrality in company with the result of normalized degree centrality, the position of Google Map, Flickr, and YouTube can be stable in spite of the slight decline of their reuse frequency. Twitter does not show the strong start as Google Maps, Flickr, or YouTube because of late entering. However, the normalized eigenvector centrality of Twitter increases constantly over time, and reaches almost the same level as the other three software services. It means that Twitter strengthens its position in the software service network.

4.2 Network Position Trends of Social Networking Services

The normalized degree centralities and the normalized betweenness centralities are calculated for the five most frequently used software services in the social networking service category. In this research, the services of the social networking category are Twitter, Facebook, Foursquare, LinkedIn, and Facebook Graph. These social networking services are chosen to investigate whether the position trends of degree centralities are due to the overall trend of a software service category and whether all services become the bridge within the software services network.

4.2.1 Degree Centrality

Figure 5 shows the trends of the normalized degree centralities of the five most frequently used services in the social networking service category. The normalized degree centralities of all services in this category grow during the study period.

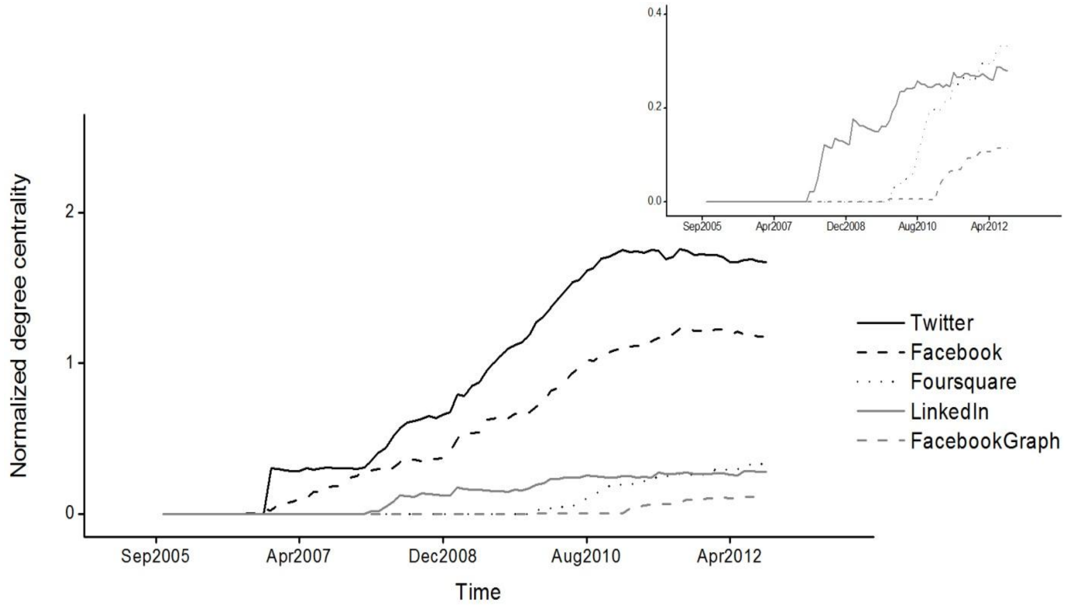

Figure 5. Trends of the normalized degree centralities for five most frequently used social networking services

In detail, the normalized degree centrality of Twitter increases fast from 0.03 in January 2008 to 1.75 in May 2011 after staying at roughly 0.03 for 13 months since its introduction in December 2006, and declines slightly until the end of the study period. The normalized degree centrality of Facebook represents almost the same pattern as that of Twitter. It consistently grows from 0.01 in August 2006 to 1.24 in October 2011, and then remains stable with a little decrease. The only difference between Twitter and Facebook is that Twitter starts to increase with a delay after entering the network while Facebook starts immediately. The other three software services increase as well. In order to see the trends clearly, the normalized degree centralities of Foursquare, LinkedIn, and

Facebook Graph are re-drawn without Twitter and Facebook (see small box in the upper right corner of Figure 5). The normalized degree centralities of Foursquare and Facebook Graph steadily grow to 0.33 and 0.12, respectively, until the end of the study period. The normalized degree centrality of LinkedIn increases with two jumps from zero to 0.13 in June 2008 and from 0.13 to 0.26 in January 2009.

The trends of the normalized degree centralities of five software services in the social networking category present that the life cycle of innovation adoption is synchronous in some services. Twitter and Facebook join the service network late, and they are in their maturity phase at the end of the study period from the perspective of three phase (i.e., ascent, maturity, and decline) innovation adoption life cycle model (Rogers, 2003).

Their life cycle is different from those of Google Maps, Flickr, and YouTube, which are in the decline phase at the end of the study period after reaching their maturity in the middle of the study period. However, it should be noted that the trends of the centralities might show differences later in their life cycle.

4.2.2 Betweenness Centrality

To determine whether the five software services are located in bridge position, the normalized betweenness centralities are calculated. The results are presented in Figure 6.

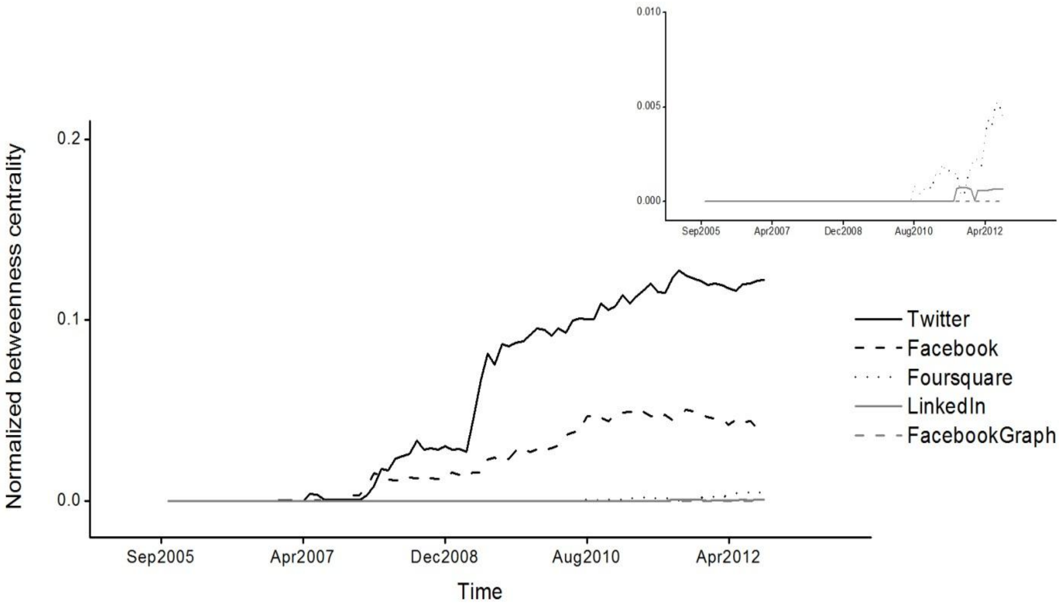

Figure 6. Trends of the normalized betweenness centralities for five most frequently used social networking services

The results of the normalized betweenness centralities show the two distinctive trends. The normalized betweenness centralities of Twitter and Facebook increase significantly. The normalized betweenness centrality of Twitter jumps from 0.03 in March 2009 to 0.08 in June 2009, and then grows to 0.12 until the end of the study period. The normalized betweenness centrality of Facebook gradually increases from 0.00 at its entrance to 0.05 in October 2011. Afterwards, it stays at about 0.04 with a slight decrease until the end of the study period. The normalized betweenness centralities of the other three software services in this category have insignificant value during the whole study period. All of the normalized betweenness centralities of Foursquare, LinkedIn, and Facebook Graph are

0.00 except for some fluctuation of Foursquare at the end of the study period (see small box in the upper right corner of Figure 6). The fluctuation value of Foursquare does not surpass 0.005.

Similar to the trends of the normalized degree centralities of the five software services in the social networking service category, the trends of the normalized betweenness centralities also show similarities, resulting in synchronous innovation adoption life cycles. They can be grouped into two characteristic trends. The normalized degree centralities and the normalized betweenness centralities of Twitter and Facebook rise to the peak at the end of the study period. They still seem to be in their ascent phase. The normalized betweenness centralities of other services in the social networking group can be negligible. Based on Figure 3 and Figure 6, Twitter and Facebook are bridges for other software services though not as strong as Google Maps. Foursquare, LinkedIn, and Facebook Graph enable innovation within clusters rather than across clusters. They approach to the core of a cluster (i.e., a set of nodes connected denser than other nodes (Scott, 1991)) rather than bridging clusters.

Looking at all results of chapter 4, the shift of the innovation trend from photo services to social networking services (as described in Figures 2 and 3) is not only due to the excellence of Twitter, but also due to the innovation trend initiated by the entire social networking service category. The top five social networking services display the strong growth in the number of connections (see Figure 5). Considering the substantial amount of joint connections that Foursquare, LinkedIn, and Facebook Graph have with Facebook and Twitter, it is speculated that Foursquare, LinkedIn, and Facebook Graph act as complementing services to the two leading social networking services. Foursquare, LinkedIn, and Facebook Graph co-develop 43, 29, and 20 mashups with Facebook or Twitter among all 92, 47, and 35 mashups that they were part of, respectively.

Chapter 5. Discussion and Conclusion

Within this thesis, the position trends of eight representative software services in the software service network are analyzed with respect to their normalized degree centralities and normalized betweenness centralities. Additionally, the position trends of four services which are the most frequently used in the entire software service network are analyzed with regard to their normalized eigenvector centrality to support the conclusion.

The results of the analysis suggest three conclusions. Firstly, the selected eight software services show life cycle behaviours of innovation adoption regarding their position within the service network (see Figures 2 and 5). Software services first grow to a maximum, remain the maximum level for a while, and then show a slight decline. For example, while all software services in the social networking service category get access to the central position throughout the study period, the most frequently used photo service, Flickr, loses its central position. The results suggest that the focus of innovation shifts from photo services to social networking services.

Secondly, the trend analysis also shows that some software services achieve hub positions in the entire network, and some software services occupy core positions within a cluster of software services (see Figures 3 and 6). Furthermore, the results of the analysis indicate that the innovation trend shifts away from a central photo service (i.e., Flickr) to social networking service (i.e., Twitter). In particular, Flickr as a central service is being replaced by Twitter in the software service network.

Thirdly, the innovation shift from a photo service to a social networking service is due not only to a single software service, but rather to the emergence of a new category of services in the software services network (see Figure 5). The reason for this relates to the fact that some of the social networking software services play the role of complements to the most frequently used social networking services, Twitter and Facebook. Together with these two social networking services, Foursquare, LinkedIn, and Facebook Graph co-develop 47%, 62%, and 57% of all mashups which they involved, respectively.

Prior research has assumed the network properties to be static, and their effect on innovation to be invariant. In this research, however, the important finding is that the network position is change according to the network evolution. The network position is not permanent anymore, even though a node has already occupied the central position such as hub. This means that existing methods and results are hardly applicable in real situations and predictable the next trend or generation without considering the dynamics. In order to make a more accurate prediction and better understanding of business relationship or other relationships, the research on the position change is needed in the field.

The findings of this thesis have important implications for academia and industry. The academic research on innovation, such as Grewal et al. (2006) or Kim et al. (2011), should also consider the evolution of the network and the changing position of node in evolving network. If a node position can change in the network, the static network property that helps demonstrate innovation performance should no longer be considered the only factor. From a managerial perspective, the software services also follow a life cycle, coherently with prior research on innovation diffusion and a life cycle of innovation adoption (Bass, 1969; Rogers, 2003). Prior research on information system network has missed the change depending on time. In addition, prior research on network analysis has described only the network structure and the evolution rule of a network (Hwang et al., 2009; Valverde and Solé, 2007). Therefore, the findings suggest that analysis method using in this thesis should be applied not only to the analysis of innovation trend in the service network, but also to innovation study more generally.

Overall, the network analysis dealt with in this thesis opens up the possibility for further studies. In order to generalize the findings, they need to be validated with a statistical method for all nodes in the software service network. Finally, the integration of this method into the decision support tool would help service owners to better understand the performance of their services.

Bibliography

Albert, R., Jeong, H. & Barabási, A.-L. (1999). Diameter of the World-Wide Web. *Nature*, 401 (6749), 130-131.

Bass, F. M. (1969). A new product growth for model consumer durables. *Management Science*, 15 (5), 215-227.

Bonacich, P. (1987). Power and centrality: A family of measures. *Am. J. of Sociol*, 92, 1170-1182.

Burt, R. S. (1992). *Structural holes: The social structure on competition*. Harvard Univ. Press, MA.

Campbell-Kelly, M. (2009). Historical reflections: The rise, fall, and resurrection of software as a service. *Communications of the ACM*, 52 (5), 28-30.

Chesbrough, H. W. (2003). *Open innovation: The new imperative for creating and profiting from technology*. Harvard Business School Press, MA.

Chesbrough, H. W. (2011). The case for open services innovation: The commodity trap. *California Management Review*, 53 (3), 5-20.

Everard, A. & Henry, R. (2002). A social network analysis of interlocked directorates in electronic commerce firms. *Electronic Commerce Research and Applications*, 1 (2), 225-234.

Freeman, L. C. (1979). Centrality in social networks: Conceptual clarification. *Social Networks*, 1, 215-239.

Girvan, M. & Newman, M. E. J. (2002). Community structures in social and biological networks. *In Proceedings of the National Academy of Science of the USA*, 99 (12), 7821-7826.

Gloor, P. A., Krauss, J., Nann, S., Fischbach, K. & Schoder, D. (2009). Web science 2.0: Identifying trends through semantic social network analysis. *In Proceedings of the 12th CSE, Vancouver, Canada.*

Granovetter, M. S. (1973). The strength of weak ties. *American Journal of Sociology*, 78 (6), 1360-1380.

Grewal, R., Lilien, G. L. & Mallapragada, G. (2006). Location, location, location: How network embeddedness affects project success in open source systems. *Management Science*, 52 (7), 1043-1056.

Haines, M. N. & Rothenberger, M. A. (2010). How a service-oriented architecture may change the software development process. *Communications of the ACM*, 53 (8), 135-140.

Hansen, M. T. (1999). The search-transfer problem: The role of weak ties in sharing knowledge across organization subunits. *Administration Science Quarterly*, 44 (1), 82-111.

Hargadon, A. B. (2002). Brokering knowledge: Linking, learning and innovation. *Research in Organizational Behavior*, 24, 41-85.

Hwang, J., Altmann, J. & Kim, K. (2009). The structural evolution of the Web2.0 service network. *Online Information Review*, 33 (6), 1040-1067.

Jin, J. H., Park, S. C. & Pyon, C. U. (2011). Finding research trend of convergence technology based on Korean R&D network. *Expert Syst. Appl.*, 38 (12), 15159-15171.

Jovanovic, B. & MacDonald, G. M. (1994). The life cycle of a competitive industry. *J. Polit. Econ.*, 102 (2), 322-347.

Kim, K., Hwang, J. & Altmann, J. (2011). Does Web2.0 foster innovation? An analysis of the openness of the Web2.0 service network. *In Proceedings of the 44th HICSS, Kauai, HI.*

Krackhardt, D. & Stern, R. N. (1980). Informal networks and organizational crises: An experimental simulation. *Social Psychology Quarterly*, 51 (2), 123-140.

Lemmens, C. E. A. V. (2004). *Innovation in Technology Alliance Networks*. Edward Elgar, MA.

Maglio, P. P., Srinivasan, S., Kreulen, J. T. & Spohrer, J. (2006). Service systems, Service scientists, SSME, and Innovation. *Communications of the ACM*, 49 (7), 81-85.

Newman, M. E. J. (2001). Clustering and preferential attachment in growing networks. *Physical Review E*, 64 (2), 025102.

Ogrinz, M. (2009). *Mashup patterns: designs and examples for the modern enterprise*. Addison-Wesley.

Opsahl, T., Agneessens, F. & Skvoretz, J. (2010). Node centrality in weighted networks: Generalizing degree and shortest paths. *Social Networks* 32 (3), 245-251.

O'Reilly (2007). What is Web2.0: design patterns and business models for the next generation of software. *Communications and Strategies*, 65, 17-37.

Papazoglou, M. P. & Georgakopoulos, D. (2003). Introduction: Service-oriented computing. *Communications of the ACM*, 46 (10), 24-28.

Rogers, E. (2003). *Diffusion of innovation*, 5th *ed*. Free Press, NY.

Sasidharan, S., Santhanam, R., Brass, D. J. & Sambamurthy, V. (2011). The effects of social network structure on enterprise systems success: A longitudinal multilevel analysis. *Information Systems Research*, 23 (3-1), 658-678.

Scott, J. (1991). *Social network analysis: A Handbook*. Sage Publication, UK.

Tsai, W. (2001). Knowledge transfer in intraorganizational networks: Effects of network position and absorptive capacity on business unit innovation and performance. *Academy of Management Journal*, 44 (5), 996-1004.

Valverde, S. & Solé, R. V. (2007). Self-organization versus hierarchy in open source social networks. *Physical Review E*, 76 (4), 046118.

Wagner, C. S., & Leydesdorff, L. (2005). Network structure, self-organization, and the growth of international collaboration in science. *Research Policy*, 34 (10), 1608-1618.

Abstract (Korean)

서비스 네트워크는 개방형 혁신 시스템으로 간주된다. 소프트웨어 공급자들이 소프트웨어를 서비스로 제공하고 사용자들이 개방형 인터페이스를 통해 서비스의 기능에 접근하는 동안, 혁신 스타일은 국부적인 혁신에서 집단적인 혁신으로 바뀌고 있다. 이러한 새로운 혁신 트렌드는 소프트웨어 서비스 네트워크 내에서 구조적, 진화적 패턴에 대한 연구들이 진행되도록 이끌어왔다. 그러나 네트워크 구조와 노드 포지션의 정적인 특성에만 관심을 기울였던 이전의 연구들은 진화하는 네트워크의 동적인 특성을 고려하지 못하고 있다. 따라서 본 연구에서는 www.programmableweb.com에 공개된 실증적 데이터를 이용하여 대표적인 소프트웨어 서비스들의 중심성을 분석함으로써 시간에 따른 네트워크 포지션의 변화를 연구한다. 연구 결과에서 각각의 서비스는 전형적인 라이프 사이클을 따르며, 혁신 트렌드는 네트워크 구조의 변화와 더불어 사진 서비스에서 소셜 네트워킹 서비스로 이동하고 있다는 사실을 보여준다. 본 연구는 혁신 연구가 정적인 네트워크 위상과 포지션뿐만 아니라, 진화하는 네트워크에서의 포지션 변화도 고려해야 한다는 것을 시사한다.

주요어 : 서비스 네트워크, Software-as-a-Service, 혁신 트렌드, 중심성, 사회연결망 분석, 복합 서비스, 개방형 혁신.
학 번 : 2012-21042

Master Thesis in Engineering

Study of Innovation Trend in Software Service Networks

- Trend Analysis based on Social Network Analysis -

소프트웨어 서비스 네트워크의 혁신 경향성 연구
: 사회 연결망 분석에 의거하여

February 2014

Graduate School of Seoul National University

Technology Management, Economics, and Policy Program

Wool rim, Lee

Study of Innovation Trend in Software Service Networks

- Trend Analysis based on Social Network Analysis -

지도교수 **Jörn Altmann**

이 논문을 공학석사학위 논문으로 제출함

2014 년 2 월

서울대학교 대학원

협동과정 기술경영경제정책 전공

이울림

이울림의 공학석사학위 논문을 인준함

2014 년 2 월

위 원 장 **Junseok Hwang** (인)

부위원장 **Jörn Altmann** (인)

위 원 **Kibae Kim** (인)

Abstract

Study of Innovation Trend in Software Service Networks

Service networks can be considered open innovation systems. As software vendors provide their software as a service (SaaS) and allow users to access its functions through open interfaces, innovation style has shifted from local innovation to collective innovation. This new innovation trend has led to the research on the structural and the evolutionary patterns in SaaS networks. However, prior research that mostly focuses on the static properties of network structure and position of nodes misses the dynamics of evolving network. In this thesis, the changes of network positions over time are studied through the investigation of the trend of the centralities of representative software services in SaaS networks. The results suggest that each software service follows a typical life cycle in its network position, and that the innovation trend shifts from the photo to social networking services by involving a transition of network structure. These results imply that innovation studies should consider not only static network topology and position, but also the changing pattern of position in evolving network.

Keywords: Service Network, Software-as-a-Service, Innovation Trend, Centralities, Social Network Analysis, Composite Services, Open Innovation.

Student Number: 2012-21042

Contents

List of Tables

List of Figures

Chapter 1. Introduction

1.1 Motivation

The advancement of IT technology enabled the provision of software services on the Internet. The demand for new business models that motivate users to innovate is underpinned by the forces that allow developers to upgrade and combine services at no charge. In this context, the network of combined software services represents an open innovation environment (Chesbrough, 2003). Examples of open innovation systems include databases of academic journals, which allow researchers to contribute their research articles, as well as read and reference the research papers by others (Wagner and Leydesdorff, 2005); the community for open source developers, which stimulates the exchange of information on software development projects (Valverde and Sole, 2007); and the network of software services with open APIs, which allows accessing service data and functions (Kim et al., 2011).

Software-as-a-Service (SaaS) is an old paradigm. Even though a computer can be distanced from other computers, users can utilize software services through the Internet. Since the emergence of cloud computing, SaaS is spotlighted again ranging from office to computer resources (Campbell-Kelly, 2009). SaaS can be served as a final good or a resource to users, and users utilize its original function and data for their innovation (Haines and Rothenberger, 2010). As software vendors provide their software as a service and allow users to access the functions via open interfaces, users can reuse the existing services for creating new services.

Because open innovation on service networks promotes the reuse of existing innovation resources, the innovation studies are interested in the structure of the entire innovation system (Chesbrough, 2011; Maglio, 2006). One of the main efforts of these studies is to apply network analysis to the network of innovation agents and resources. It follows the research on social networks (Freeman, 1979), statistical physics (Albert et al., 1999), and co-authorship networks (Newman, 2001; Wagner and Leydesdorff, 2005). The new trend of innovation, in which each service is reused by other services, invites researchers on information systems to analyze the structure and the evolution of these information systems as a platform for “open innovation” (Chesbrough, 2003).

1.2 Problem Description

An important part in prior innovation research on networks has been to investigate the structure and the evolutionary pattern of networks (Newman, 2001; Valverde and Sole, 2007; Wagner and Leydesdorff, 2005) and to correlate the network characteristics with innovation performance (Granovetter, 1973; Grewal et al., 2006; Krackhardt and Stern, 1980). The innovation research examined the relationship between a node’s social network position in a network structure and its innovation performance (Granovetter, 1973; Krackhardt and Stern, 1980). Similar analyses have been performed for innovation communities (Grewal et al., 2006; Valverde and Sole, 2007) and innovation resources (Kim et al., 2011).

However, previous studies have considered the network properties as static, and their effect on innovation as invariant. In this sense, these studies miss the complex and dynamic behaviour of each node in an evolving network. This thesis complements prior research in that it explores the dynamic behaviour of an innovation system by analyzing the position trends of representative software services in the evolving SaaS network depending on time.

1.3 Research Goal

The main objectives of this thesis are to introduce the dynamic behavior into the innovation studies, to investigate the trend of centralities of representative software services in a SaaS network, and to discuss the patterns of changing positions of software services in the evolving SaaS network.

1.4 Research Questions

Two main questions are adduced to achieve the research goal: (1) Is the position of a node fixed in a network? (2) What does a position change imply for innovation? To answer these two main questions, the following four sub-questions (two per each of the main questions) need to be addressed, respectively: (1a) Which software services locate at the central position? (1b) Do software services at the central position shift? (2a) Which service is more appropriate for innovation in the evolving network? (2b) Is the innovation trend of a service related to service category?

1.5 Research Outline

The remainder of this thesis is organized as follows. The next chapter gives a brief description of the conceptual background of SaaS innovation systems and the network perspectives of innovation studies.

Chapter 3 describes the collected data and how to form the SaaS network with the data, and explains how to choose representative software services and measure the centralities.

The software service network considered in this study has been built from the empirical data surveyed on the public Web site www.programmableweb.com. This network consists of nodes and links. Each node represents a software service, and each link denotes the joint use of two existing software services for the development of new software services. Software services with open application programming interfaces (APIs) can be used for creating new composite software services. The representative software services have been selected from 7427 software services according to the frequency of their use throughout the study period.

The normalized degree centrality, the normalized betweenness centrality, and the normalized eigenvector centrality of the representative software services have been measured to examine their position over time. These results allow examining whether the innovation leader shifts from one service to another. The measurement of the normalized degree centrality and the normalized betweenness centrality for the four most frequently used software services in the social networking service category has been added. It can help reveal whether the shift of an innovation leader’s position is related to the innovation of a group of services (i.e., a category).

Chapter 4 presents the results of the analysis, specifically, the trend analysis of nodes' positions of the representative software services within the SaaS network. The main results of the research are as follows. First, similarly to the innovation adoption life cycle (Rogers, 2003), each software service exhibits a life cycle including the growing, maturity, and decline phase. Second, some software services achieve the hub positions in the entire network, and some software services achieve the core positions within a cluster of software services. It can be noted that the photo service (i.e., Flickr) has been replaced by a social networking service (i.e., Twitter), while the mapping service (i.e., Google Maps) maintains its hub position in the network. Finally, the innovation adoption life cycle of a software service depends on the innovation trend of its service category. The reason for this can be found in the fact that some of the software services in the same category act as complements to the most frequently used services, pushing the success of those most frequently used services even further.

The findings of this study have entrepreneurial implications with regard to the positioning strategy of software service vendors. Furthermore, these are also expected to redirect the network analysis of an innovation system from a static perspective to a dynamic perspective. The conclusions and implications of this study are explained in Chapter 5.

Chapter 2. Literature Review

2.1 Open Innovation in Software Service Network

As IT technology advanced and new business models emerged that motivate consumer to participate in an innovation process (e.g., Web 2.0 service creation), software vendors offered their software as a service (SaaS). SaaS is a paradigm, which allows users to run software that is installed remotely via the Internet. The SaaS paradigm emerged with commercial computing in the 1980s but has been in the downturn with the growth of personal computer. Since the emergence of cloud computing, SaaS is spotlighted again in a various areas ranging from office software (e.g., Google docs) to computing resources (e.g., Amazon S3) (Campbell-Kelly, 2009).

One type of SaaS implementation is based on the Web services under the concept of Service Oriented Architecture (SOA). SOA defines how users can compose and reuse services through the open interfaces. It is called open Application Programming Interface (API), and one of the famous example is Google Maps (Haines and Rothenberger, 2010; Papazoglou and Georgakopoulos, 2003). A composite service is created by adding a unique value to an existing software service or by integrating several existing software services. This composite service is called “mashup” (Ogrinz, 2009).

Figure 1 is an example of a composite service, Weather Bonk. It provides weather information on a map by accessing open APIs of Google Maps and Weather Bug. According to the definition of software service network, Weather Bonk service connects all eight software services with each other (Figure 1), resulting in a fully connected graph.

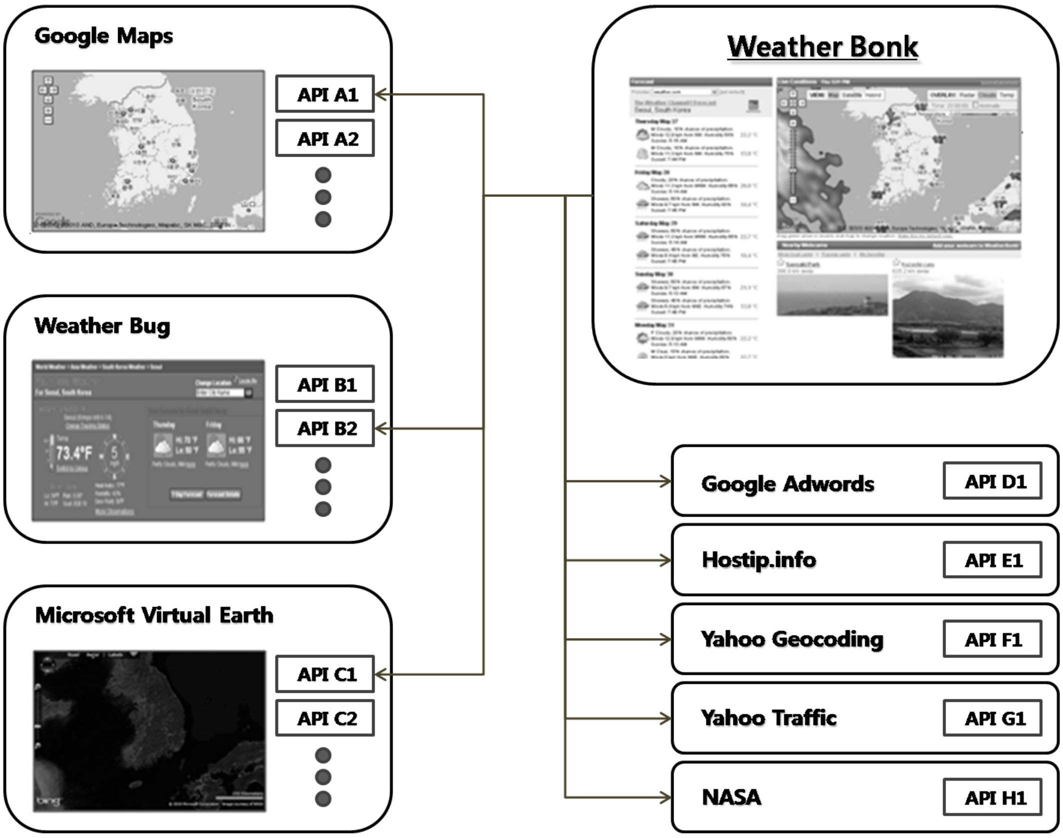

Figure 1. Example of a composite service, Weather Bonk

In this architecture, software vendors achieve a huge scale of innovation utilizing their service users (O'Reilly, 2007). This is open innovation in a software service environment. Innovation occurs through the free sharing of software services with their service users and even with their competitors (Chesbrough, 2003). Any software service provider can participate in the innovation by simply opening up the APIs. This means that users can reuse the data and functions of those services for their own service development, any users can join in this kind of innovation. This is the reason why the innovation system grows with a specific structure.

2.2 Network Position and Innovation

In the social network studies, a diversity of measures has been developed to analyze the position of nodes (agents) in a network. The position of an agent in a network is important, because the position has a relationship with the role of the agent in the society (Scott, 1991). The most popular measure is centrality based on the degree of a node or the geodesic between two nodes (Freeman, 1979). The degree centrality of node is measured by the number of adjacent nodes, which implies how deeply a node is embedded among its neighbours. The betweenness centrality of a node considers the number of shortest paths (geodesics) through the node compared to all possible geodesics. It is used to determine the extent to which the node is interconnected with other nodes. If a node has high betweenness centrality and low degree centrality, it can be supposed that the node might bridge several separated communities within the network (Everard and Henry, 2002). In addition, it is also important to understand which nodes are connected. As a node's power in a network is determined by the power of its neighbors (Bonacich, 1987), the eigenvector centrality has been defined. The eigenvector centrality indicates the sum of the degree centralities of the adjacent nodes.

Empirical research about the network reveals that the degree distribution of node in some real networks shows the scale-free property. The scale-free property represents that the frequency of node degree decays by the power function. This distribution is very inhomogeneous, or highly skewed, compared with the exponential distribution of random networks or the distribution of regular networks (Albert et al, 1999). In a scale-free network, few nodes are connected with a majority of nodes in order to make the distance between nodes small, while the degree of most nodes is low. The few nodes with high node degree are called "hubs".

Since innovation performance of an agent is closely related to its network position, the heterogeneity of network position is important for the study of innovation in the network (Burt, 1992; Granovetter, 1973; Grewal et al., 2006). As the network position of an agent can be quite different, the performance of an agent can vary considerably. Innovation is a recombining process of fragmented existing knowledge (Hargadon 2002). An agent (e.g., a researcher or a firm) can create new knowledge by recombining knowledge of the group. If an agent bridges two distant and separate clusters within the network through (even) low and infrequent connections with agents which belong to these clusters, the agent enables an innovation. The agent can access diverse knowledge efficiently and effectively so that new ideas can emerge from the whole system, not a clustered group only (Burt, 1992; Granovetter, 1973).

Prior research on the analysis of innovation in the network overlooks the dynamic aspect of network evolution. It just concentrates on static property and statistical relationship between nodes. For instance, in the empirical network analysis and the evolutionary model introduced by Albert et al. (1999), the characteristics of scale-free networks are invariant and the network is formed by a preferential attachment rule that determines the link between a new node and existing nodes. According to this model, a node is located at the central position forever, if it has been selected as a hub at the early stage.

Innovation research exploring the effect of network position on innovation also assumes that the network structure does not change in the evolutionary network. Granovetter (1973) and Burt (1992) emphasized the importance of nodes which are connected with separate and distant clusters for innovation under the fixed network structure. The network position affecting innovation has been investigated in various conditions specifying the relationship between network position and innovation performance. For example, the more central a node (agent) in a network is, the higher the innovation performance of the node (agent) is under the conditions that the node has a good absorptive capacity (Tsai, 2001), that the node is in a central group (Sasidharan et al., 2011), and that the node has weak ties (Hansen, 1999).

In these literatures, the analyses were carried out with the data which represent snapshot of network structure and innovation performance. These analyses of Granovetter (1973), Burt (1992), Tsai (2001), Sasidharan et al. (2011), and Hansen (1999) did not consider the time factor in their discussion concerning the relationship between network position and innovation.

Innovation systems are more dynamic than the network studies mentioned. On the one hand, innovation systems of diverse levels (e.g., a single technology or entire industry) show life cycle behaviour from emergence to decline through maturity. These are described by the models that explain the driving force of the rise and fall of an innovation system through the diffusion of technology (Bass, 1969; Rogers, 2003), the opportunity caused by a new technology and the competition (Jovanovic and MacDonald, 1994), and the opportunity of strategic alliances with competitors (Lemmens, 2004). On the other hand, especially in case of innovation through collective intelligence, the trend of innovation varies as the interest of the crowd shifts (Jin et al., 2009). The changing interest of the crowd is often revealed in the service networks including Web sites, blogs, and some special forums, and the analysis of the changing interest can be used to predict the innovation trend (Gloor et al., 2009).

As a result of these perspectives, the question rises as to whether the structure of an innovation network remains stable while the internal status of a node in the network changes. One of the research consequences about the question shows that the hubs in the software service network are replaced while the scale-free topology is kept (Hwang et al., 2009).

Chapter 3. Methodology

3.1 Service Network

The data has been gathered from the Web site www.programmableweb.com which lists information about software services. The information includes the name, the launch date, and the brief description of the services, the openness of the service's APIs, and the services that reuse the existing services. If a software service is composed, this software service is called mashup in Web site. The information of all software services has been collected since the first composite service was added on September 14th, 2005. The last data set for the research is September 30th, 2012.

The software service network is defined as a set of nodes and links between these nodes. Each node represents a software service that opens up its APIs, and a link indicates the existence of a composite service that uses the connected nodes. The creation of a composite service yields a complete graph of software services that is used for the development of the composite service. For instance, shown in Figure 1, the creation of Weather Bonk generates 28 links between the eight software services with open APIs (Google Maps, Microsoft Virtual Earth, Weather Bug, Google Adwords, hostip.info, Yahoo Geocoding, Yahoo Traffic, and NASA), connecting all eight nodes with each other.

The links of the software service network are non-directional and weighted. That is, a link does not show the information of the source and destination of its relationship, but displays the frequency of use of the link. To illustrate, assuming that a new composite service is created using Google Maps and Yahoo Traffic in addition to the SaaS network including the Weather Bonk, the weight of the link between Google Maps and Yahoo Traffic is 2 while the weight of other links is still 1.

3.2 Measures

While the software service network is a weighted graph, the centrality measures have usually been defined for binary graphs (Bonacich, 1987; Everard and Henry, 2002; Freeman, 1979). Therefore, it is necessary to modify the centralities for weighted graphs (Opsahl et al., 2010). Following terminologies are introduced in this thesis. Let w_{ij} be the weight of the link between nodes i and j belonging to the network G whose size is g. The weight of a link in a SaaS network denotes the occurrence frequency of collaboration between the two nodes, and w_{ij} is an integer in a SaaS network. If w_{ij} is zero, there is no link between the two nodes. Therefore, the weight represents the frequency of interaction between the nodes.

In a weighted graph, the degree centrality is defined as the sum of the weights of the links that a node has. That is, the degree centrality of node i in a weighted graph is the sum of weights of the links w_{ij} that node i has with node j. The degree centrality of a node in a weighted graph is also called the strength of the node because it illustrates the strength of connectivity of the node (Opsahl et al., 2010). The degree centrality seems to be increased as the network size increases even in the networks with identical density. In order to eliminate the effect of network size from degree centrality, the degree centrality of binary graph is normalized by the maximum possible number of links that a node can have. The normalized degree centrality in a binary graph varies between *0* and *1*. If each node is linked with all the other nodes, the degree centrality goes to *1*. If a node is not connected with any other nodes, the degree centrality goes to *0*. The degree centrality of weighted graph is also normalized by the maximum possible number of neighbours that a node can have (i.e., $g-1$):

$$s'(i) = CD'(i) = \sum_{j \neq i} w_{ij} / (g-1). \qquad \text{Eq. (1)}$$

However, applying the normalized degree centrality to a weighted graph has a limitation. If the weight is larger than *1*, the normalized degree centrality can become larger than *1*. In spite of that, the normalized degree centrality is a good measure to compare the centralities of nodes in the different networks of various sizes.

The betweenness centrality is defined on the basis of the shortest paths (geodesics). In this thesis, the betweenness centrality for a weighted graph is employed instead of the betweenness centrality for a binary graph in common with the degree centrality. Thus, the shortest path length in a weighted graph should be re-defined. In general, the shortest path between two nodes is the path passing the smallest number of links between them, and the shortest path length is the number of links of the shortest path. In a weighted graph, a pair of nodes is said to be closer as the weight of link between nodes gets larger in the software service network. For this reason, it is reasonable to assume that between any two nodes the path length considered the weight is smaller than the path length without considering. Therefore, the shortest path length $d^w(i,j)$ between two nodes i and j in a weighted graph G is determined as the sum of the inversed weights of the links on the shortest path between the nodes (Opsahl et al., 2010):

$$d^w(i,j) = \min \{|p_{ij}| \text{ } for\ all\ p_{ij}\ where\ |p_{ij}| = 1/w_{ih} + \cdots + 1/w_{kj}\}. \qquad \text{Eq. (2)}$$

Let $\sigma_{ij}{}^{w}$ be the number of shortest paths between node i and j in a weighted graph G, and $\sigma_{ij}{}^{w}(v)$ be the number of shortest paths between node i and j passing through the node v. The betweenness centrality of node v in a weighted graph is defined as the sum of $\sigma_{ij}{}^{w}(v)$ / $\sigma_{ij}{}^{w}$ for all nodes i and j in the same manner as the betweenness centrality in a binary graph (Freeman, 1979). Since the number of pairs of nodes increases as the network size increases, the betweenness centrality needs to be normalized by the maximum possible number of pairs of any two nodes except for the node v in a weighted graph G with size g, $(g\text{-}1)\cdot(g\text{-}2)/2$. Therefore, the normalized betweenness centrality for a weighted graph is defined as:

$$\mathrm{CB}'(i) = \sum_{i,j \neq i} \sigma_{ij}{}^{w}(v) / \sigma_{ij}{}^{w} / ((g\text{-}1)\cdot(g\text{-}2) / 2). \qquad \text{Eq. (3)}$$

The normalized betweenness centralities for a binary graph and a weighted graph vary between 0 and 1. If the shortest path passing through the node does not exist, the normalized betweenness centrality of the node is *0*. If all the shortest paths pass through the node, the normalized betweenness centrality of the node is *1*.

Lastly, the definition of eigenvector centrality is needed. The eigenvector centrality of a node is the sum of the eigenvector centralities of its adjacent nodes. The adjacency of each node in a binary graph whose network size is g is represented as a g by g matrix $\boldsymbol{A}$. If node i and node j are directly connected, the element of matrix $\boldsymbol{A}$, a_{ij}, is *1*, and if not, the element of matrix $\boldsymbol{A}$ is *0*.

In this thesis, $\boldsymbol{x}$ is the vector of the eigenvector centralities of all nodes in the graph, which should satisfy the condition $\boldsymbol{A}\ \boldsymbol{x} = \lambda\ \boldsymbol{x}$. The symbol λ indicates an eigenvalue (Bonacich, 1987). The eigenvector centrality is normalized by the vector length $||\boldsymbol{x}||$, and the eigenvector centrality of a node in a weighted graph is the sum of the eigenvector centralities of all its adjacent nodes. Let $\boldsymbol{W}$ be a g by g matrix where element w_{ij} is the weight between nodes i and node j in a graph G with size g. The normalized eigenvector centrality of node i is defined as the i-th element of $\boldsymbol{x}$, satisfying the following condition:

$$\mathbf{W}\,\boldsymbol{x} = \lambda\,\boldsymbol{x}. \qquad \text{Eq. (4)}$$

The number of eigenvalues is g at most. The eigenvector with the largest eigenvalue among the multiple eigenvalues is selected because it involves the nodes in the main component that consists of the largest amount of nodes segregated from the other components. And the eigenvector centralities of all nodes which are not belong to the largest component are zero.

Chapter 4. Analysis Results

The data set has been surveyed from www.programmableweb.com which lists information about software services. The information includes the name, the launch date, and the brief description of the services, the accessibility of the APIs, and the services used for creating the composite services. The data have been collected from September 14th, 2005 when the first composite service was added to September 30th, 2012. In this period, 7427 software services have been registered, 6780 services are composite services among the registered software services. The composite services were developed by utilizing 1153 software services that offered their functionalities through the open APIs. From the surveyed data, a software service network for the analysis consists of 1153 nodes and 23573 links. This network is undirected and weighted, and the weight of the link between a pair of nodes represents the number of composed software services based on these two nodes.

For the analysis, eight representative software services are selected according to the frequency of use during the study period. Among them, four services (i.e., Google Maps, Twitter, YouTube, and Flickr) are the most frequently used services in the software service network, and the remaining four services (i.e., Facebook, Foursquare, LinkedIn, and Facebook Graph) and Twitter are the most frequently used services in the social networking service category. The social networking service category is chosen because it becomes popular later than the other service categories such as mapping, video, and photo.

The software services in the social networking service category are expected to show the different trends in their position compared with the four most frequently used software services. Before measuring the centralities, the brief introduction about selected eight representative software services is indicated in Table 1.

Table 1. Description of the selected software services

Service Name	Provider	Service Category	Entrance Time	Number of Uses
Google Maps	Google	Mapping	September 2005	2263
Twitter	Twitter	Social Networking	January 2007	644
YouTube	Google	Video	April 2006	598
Flickr	Yahoo	Photo	September 2005	590
Facebook	Facebook	Social Networking	August 2006	352
Foursquare	Foursquare	Social Networking	January 2010	82
LinkedIn	LinkedIn	Social Networking	February 2008	46
Facebook Graph	Facebook	Social Networking	December 2009	37

The research objective is to investigate the change of network position over time. The remainder of this section demonstrates the analysis results of the network position trends concerning the selected eight software services. The network positions of all eight services are determined by the normalized degree centrality and the normalized betweenness centrality. In addition, the eigenvector centralities of four services which are the most frequently used in the software service network are measured in order to complementing the analysis result. However, the eigenvector centrality of the social networking service category is ignored in this chapter because it does not show any notable trends.

The software service network in this research is unidirectional and weighted. The centralities are measured monthly, and the starting point of all services is September 2005. For example, the service network of May 2011 consists of all nodes and links from September 2005 to May 2011.

4.1 Network Position Trends of Most Frequently Used Services

4.1.1 Degree Centrality

The normalized degree centralities of four most frequently used software services (Google Maps, Twitter, Flickr, and YouTube) are measured respectively, in order to investigate how deeply each selected service is embedded among its neighbouring nodes in the software service network and how the embeddedness changes depending on time.

Figure 2 illustrates the trends of the normalized degree centralities during the study period and shows that the trends of four representative services shine out compared to the average normalized degree centrality of all other services (i.e., total) in the software service network. The normalized degree centralities of Google Maps, Flickr, and

YouTube show a similar tendency. Each of these services increases fast at the early period, and then decline until the end of the study period after staying at a certain level (i.e., prosperity period). The normalized degree centrality of Google Maps approximately increases from 0.65 in September 2005 to 2.70 in December 2010, but the growing rate is decreasing greatly since December 2007. After December 2010, the normalized degree centrality declines from 2.70 to 2.28 at the end of the study period. Likewise, the normalized degree centrality of Flickr soars from 0.10 in September 2005 to 2.01 in December 2007, and the normalized degree centrality of YouTube rises from 0.14 in January 2006 to 1.85 in September 2008. After the short period of slight increase, the normalized degree centralities of Flickr and YouTube decline from 2.18 in May 2010 to 1.68 and from 2.02 in January 2011 to 1.75 until the end of study period, respectively. Twitter is introduced later than the other three software services. The normalized degree centrality of Twitter remains stable at about 0.30 until January 2008, and grows fast to 1.75 in May 2011. After growing, it stays stable again with a slight decrease until the end of study period.

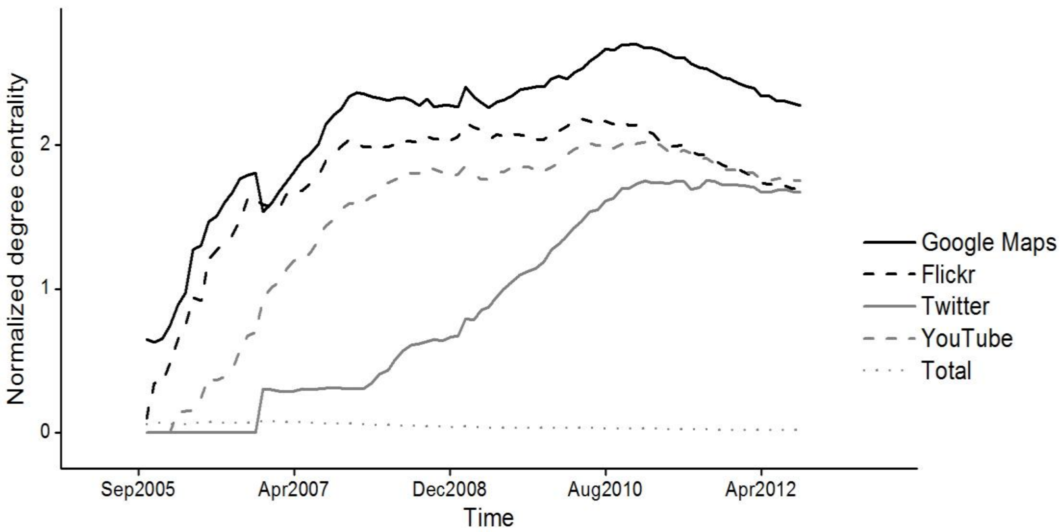

Figure 2. Trends of the normalized degree centralities of the four most frequently used software services

The trends of normalized degree centralities of the selected software services look like a life cycle of technology. Prior research on technology diffusion or user's adoption of technology suggests that a new technology is adopted exponentially after an inactive early period and then slows down, so that the cumulated adoption rate of the technology shows an S-like curve (Bass, 1969; Rogers, 2003). For the software service network, if a software service is frequently reused for developing the composite service, this service reaches the central position. But if a software service is not reused anymore or frequently, this service is pushed out to the periphery of the network. From this perspective, Google Maps, Flickr, and YouTube have grown to their maturity phase during the study period, and now face their saturation period. Twitter is in the maturity phase at the end of the study period because Twitter gets in to the software service network late.

In this research, Google Maps, Flickr, and YouTube do not show an early inactive period. This tendency might be related to the fact that these three services enter the network at the beginning of the evolution of service network. At that point, these services are already known to users, and users already know how to utilize these services. Therefore, these services do not need to wait for being diffused by "imitators" (Bass, 1969). However, Twitter displays an early inactive period after entering the service network. Users have to learn the value of Twitter as it is not the popular service at the early stage.

4.1.2 Betweenness Centrality

The normalized betweenness centralities are calculated to find out whether the four representative software services are bridges for other software services in the software service network, and to analyze the trends of the betweenness centralities of the selected services. The results of the normalized betweenness centralities are described in Figure 3. This shows that the trend of each software service is idiosyncratic. The normalized betweenness centrality of Google Maps is substantially higher than the other three services through the whole study period. It remains at about 0.22 on average after a slight drop. The normalized betweenness centrality of Flickr goes down slowly after the peak of the initial period. It rises rapidly from 0.00 in November 2005 to 0.18 in June 2006, and then decreases to 0.07 at the end of the study period with some fluctuations. The normalized betweenness centrality of YouTube increases gradually during the first half period, and keep the same level during the second half period. To be more specific, it grows from 0.00 in April 2006 to 0.09 in October 2009, and stays at this level after that. The normalized betweenness centrality of Twitter has another interesting trend. It remains 0.00 between December 2006 and January 2008, and jumps to 0.03 in July 2008. It jumps again from 0.03 in March 2009 to 0.08 in June 2009, and then gradually increases to 0.12 by the end of study period.

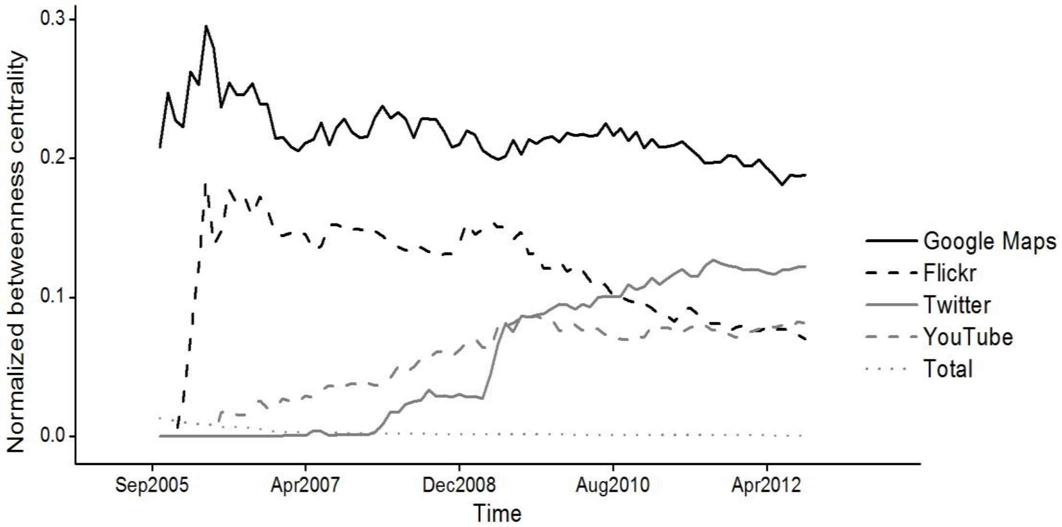

Figure 3. Trends of the normalized betweenness centralities of the four most frequently used software services

The shift of innovation trend can be demonstrated by the trend of normalized betweenness centrality. A node with high betweenness centrality and low degree centrality implies that the node bridges multiple separate and distant clusters, while a node with high betweenness centrality and high degree centralities occupies the hub position (Everard and Henry, 2002). In this software service network, Google Maps and Flickr are the hubs initially because they are connected to a lot of nodes and clusters in the network. Over time, normalized betweenness centrality of Flickr is declined while Google Maps keeps its level. Twitter increases in both normalized degree centrality and normalized betweenness centrality. From August 2010, Twitter even becomes a more important hub than Flickr. Flickr loses its initial hub position and becomes at the same level as YouTube in September 2012. Twitter, as a social networking service, could interconnect many more clusters of services than Flickr could. It is also notable that the normalized betweenness centrality of YouTube grows late while its normalized degree centrality is already high in the early periods. This means that YouTube is linked with many nodes but not connected with many clusters. In a graph theoretical perspective, it can be stated that YouTube is a core in some clusters, not a hub in the global network.

Considering the results of both normalized degree centrality and normalized betweenness centrality, Google Maps is expected to maintain its structural position for a considerable period of time. Innovation will happen through the use of Google Maps together with other major services (e.g., YouTube, Flickr, or Twitter). However, not all services can keep their position. As the normalized betweenness centrality of Flickr represents, it is likely that Flickr will not be able to keep the position of a major player in the near future. Flickr loses its attraction. The place left by Flickr will be substituted with

Twitter, because both normalized degree centrality and normalized betweenness centrality of Twitter increase gradually. It leads to the creation of new composite services. Therefore, the result is summarized that the innovation trend shifts from a photo service (i.e., Flickr) to a social networking service (i.e., Twitter).

4.1.3 Eigenvector Centrality

The eigenvalues and the eigenvectors of four representative software services are calculated, and the eigenvector which has the largest eigenvalue is select. The results are classified as three types of trends: shock, fast growth, and late growth. The normalized eigenvector centralities of Google Maps and Flickr start with high value (shock). Excluding the early fluctuation, the normalized eigenvector centralities of Google Maps and Flickr are about 0.44 and 0.42 on average during the study period. The normalized eigenvector centrality of YouTube grows fast during the initial period (fast growth). It increases from 0.08 in January 2006 to 0.42 in October 2008, and remains at 0.41 on average until the end of the study period. The normalized eigenvector centrality of Twitter goes up slowly compared to the other three services (late growth). It rises from 0.06 in December 2006 to 0.34 in September 2012.

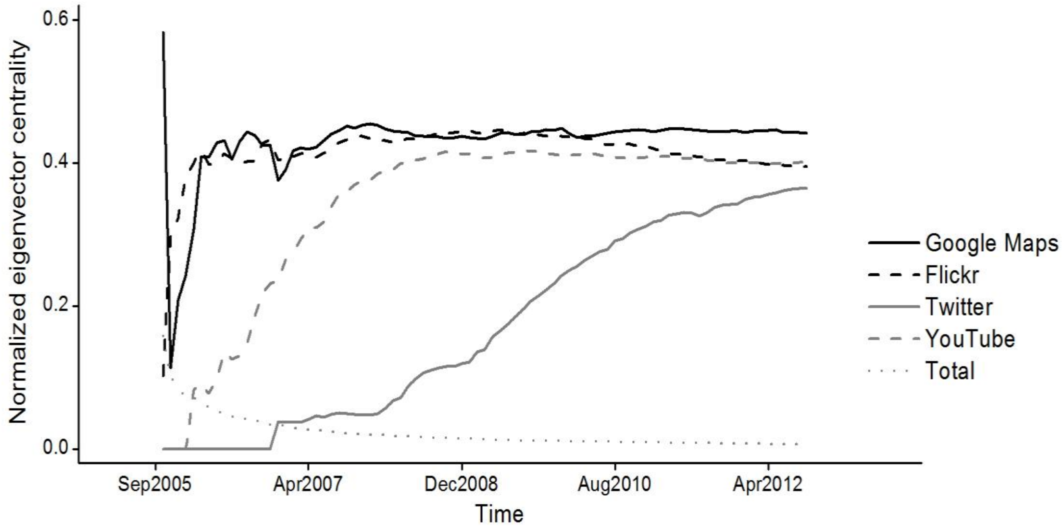

Figure 4. Trends of the normalized eigenvector centralities of the four most frequently used software services

The trends of normalized eigenvector centralities of the selected software services imply that the innovation in the software service network evolves in a complicate manner. As the normalized eigenvector centrality of a node indicates the strength of neighbouring nodes, it can be said that Google Maps, Flickr, and YouTube are stably linked with many and powerful nodes. Considering the result of normalized eigenvector centrality in company with the result of normalized degree centrality, the position of Google Map, Flickr, and YouTube can be stable in spite of the slight decline of their reuse frequency. Twitter does not show the strong start as Google Maps, Flickr, or YouTube because of late entering. However, the normalized eigenvector centrality of Twitter increases constantly over time, and reaches almost the same level as the other three software services. It means that Twitter strengthens its position in the software service network.

4.2 Network Position Trends of Social Networking Services

The normalized degree centralities and the normalized betweenness centralities are calculated for the five most frequently used software services in the social networking service category. In this research, the services of the social networking category are Twitter, Facebook, Foursquare, LinkedIn, and Facebook Graph. These social networking services are chosen to investigate whether the position trends of degree centralities are due to the overall trend of a software service category and whether all services become the bridge within the software services network.

4.2.1 Degree Centrality

Figure 5 shows the trends of the normalized degree centralities of the five most frequently used services in the social networking service category. The normalized degree centralities of all services in this category grow during the study period.

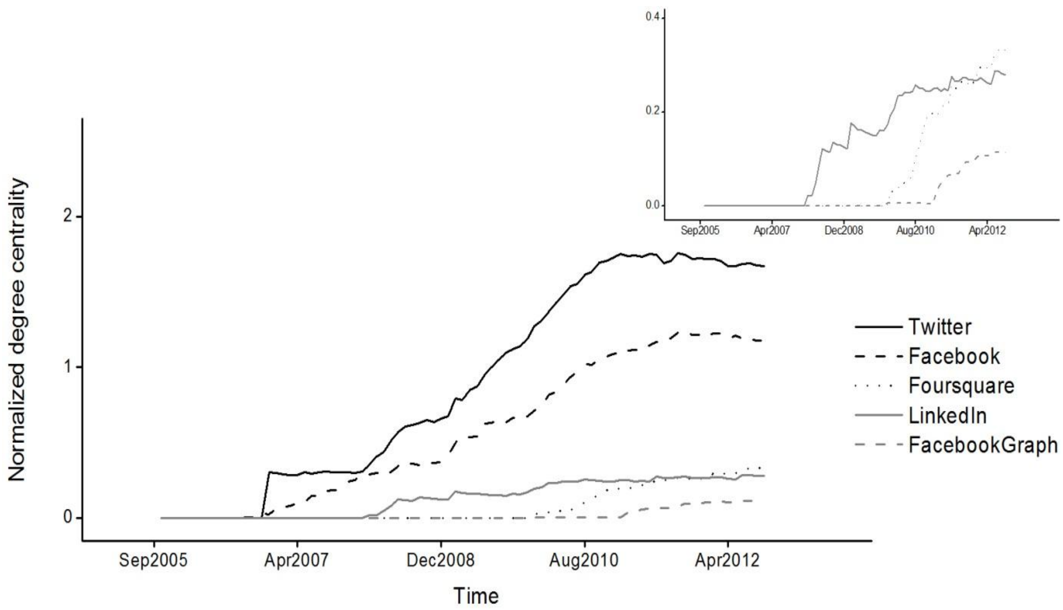

Figure 5. Trends of the normalized degree centralities for five most frequently used social networking services

In detail, the normalized degree centrality of Twitter increases fast from 0.03 in January 2008 to 1.75 in May 2011 after staying at roughly 0.03 for 13 months since its introduction in December 2006, and declines slightly until the end of the study period. The normalized degree centrality of Facebook represents almost the same pattern as that of Twitter. It consistently grows from 0.01 in August 2006 to 1.24 in October 2011, and then remains stable with a little decrease. The only difference between Twitter and Facebook is that Twitter starts to increase with a delay after entering the network while Facebook starts immediately. The other three software services increase as well. In order to see the trends clearly, the normalized degree centralities of Foursquare, LinkedIn, and

Facebook Graph are re-drawn without Twitter and Facebook (see small box in the upper right corner of Figure 5). The normalized degree centralities of Foursquare and Facebook Graph steadily grow to 0.33 and 0.12, respectively, until the end of the study period. The normalized degree centrality of LinkedIn increases with two jumps from zero to 0.13 in June 2008 and from 0.13 to 0.26 in January 2009.

The trends of the normalized degree centralities of five software services in the social networking category present that the life cycle of innovation adoption is synchronous in some services. Twitter and Facebook join the service network late, and they are in their maturity phase at the end of the study period from the perspective of three phase (i.e., ascent, maturity, and decline) innovation adoption life cycle model (Rogers, 2003).

Their life cycle is different from those of Google Maps, Flickr, and YouTube, which are in the decline phase at the end of the study period after reaching their maturity in the middle of the study period. However, it should be noted that the trends of the centralities might show differences later in their life cycle.

4.2.2 Betweenness Centrality

To determine whether the five software services are located in bridge position, the normalized betweenness centralities are calculated. The results are presented in Figure 6.

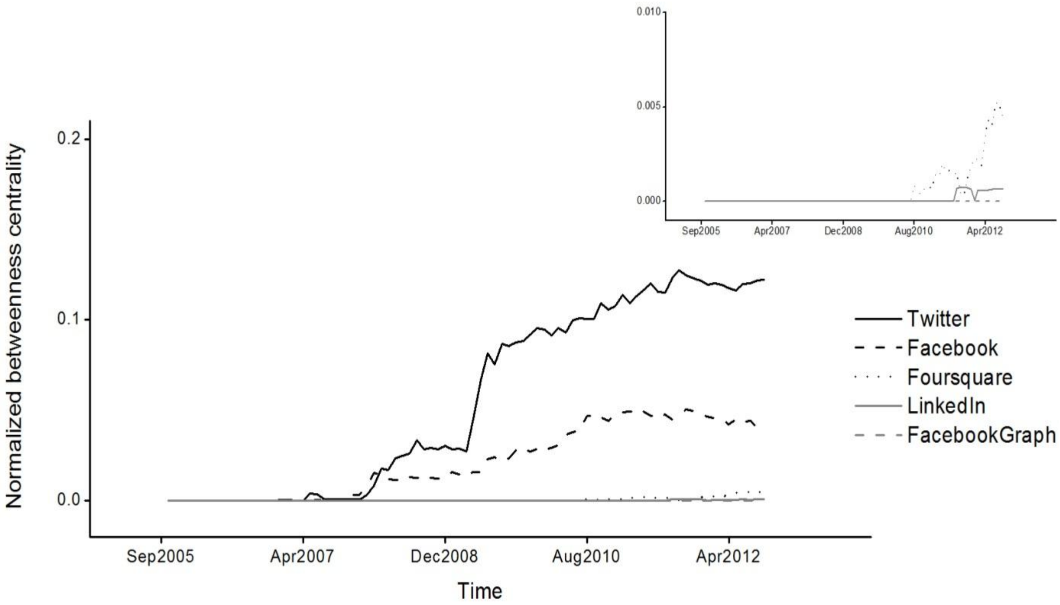

Figure 6. Trends of the normalized betweenness centralities for five most frequently used social networking services

The results of the normalized betweenness centralities show the two distinctive trends. The normalized betweenness centralities of Twitter and Facebook increase significantly. The normalized betweenness centrality of Twitter jumps from 0.03 in March 2009 to 0.08 in June 2009, and then grows to 0.12 until the end of the study period. The normalized betweenness centrality of Facebook gradually increases from 0.00 at its entrance to 0.05 in October 2011. Afterwards, it stays at about 0.04 with a slight decrease until the end of the study period. The normalized betweenness centralities of the other three software services in this category have insignificant value during the whole study period. All of the normalized betweenness centralities of Foursquare, LinkedIn, and Facebook Graph are

0.00 except for some fluctuation of Foursquare at the end of the study period (see small box in the upper right corner of Figure 6). The fluctuation value of Foursquare does not surpass 0.005.

Similar to the trends of the normalized degree centralities of the five software services in the social networking service category, the trends of the normalized betweenness centralities also show similarities, resulting in synchronous innovation adoption life cycles. They can be grouped into two characteristic trends. The normalized degree centralities and the normalized betweenness centralities of Twitter and Facebook rise to the peak at the end of the study period. They still seem to be in their ascent phase. The normalized betweenness centralities of other services in the social networking group can be negligible. Based on Figure 3 and Figure 6, Twitter and Facebook are bridges for other software services though not as strong as Google Maps. Foursquare, LinkedIn, and Facebook Graph enable innovation within clusters rather than across clusters. They approach to the core of a cluster (i.e., a set of nodes connected denser than other nodes (Scott, 1991)) rather than bridging clusters.

Looking at all results of chapter 4, the shift of the innovation trend from photo services to social networking services (as described in Figures 2 and 3) is not only due to the excellence of Twitter, but also due to the innovation trend initiated by the entire social networking service category. The top five social networking services display the strong growth in the number of connections (see Figure 5). Considering the substantial amount of joint connections that Foursquare, LinkedIn, and Facebook Graph have with Facebook and Twitter, it is speculated that Foursquare, LinkedIn, and Facebook Graph act as complementing services to the two leading social networking services. Foursquare, LinkedIn, and Facebook Graph co-develop 43, 29, and 20 mashups with Facebook or Twitter among all 92, 47, and 35 mashups that they were part of, respectively.

Chapter 5. Discussion and Conclusion

Within this thesis, the position trends of eight representative software services in the software service network are analyzed with respect to their normalized degree centralities and normalized betweenness centralities. Additionally, the position trends of four services which are the most frequently used in the entire software service network are analyzed with regard to their normalized eigenvector centrality to support the conclusion.

The results of the analysis suggest three conclusions. Firstly, the selected eight software services show life cycle behaviours of innovation adoption regarding their position within the service network (see Figures 2 and 5). Software services first grow to a maximum, remain the maximum level for a while, and then show a slight decline. For example, while all software services in the social networking service category get access to the central position throughout the study period, the most frequently used photo service, Flickr, loses its central position. The results suggest that the focus of innovation shifts from photo services to social networking services.

Secondly, the trend analysis also shows that some software services achieve hub positions in the entire network, and some software services occupy core positions within a cluster of software services (see Figures 3 and 6). Furthermore, the results of the analysis indicate that the innovation trend shifts away from a central photo service (i.e., Flickr) to social networking service (i.e., Twitter). In particular, Flickr as a central service is being replaced by Twitter in the software service network.

Thirdly, the innovation shift from a photo service to a social networking service is due not only to a single software service, but rather to the emergence of a new category of services in the software services network (see Figure 5). The reason for this relates to the fact that some of the social networking software services play the role of complements to the most frequently used social networking services, Twitter and Facebook. Together with these two social networking services, Foursquare, LinkedIn, and Facebook Graph co-develop 47%, 62%, and 57% of all mashups which they involved, respectively.

Prior research has assumed the network properties to be static, and their effect on innovation to be invariant. In this research, however, the important finding is that the network position is change according to the network evolution. The network position is not permanent anymore, even though a node has already occupied the central position such as hub. This means that existing methods and results are hardly applicable in real situations and predictable the next trend or generation without considering the dynamics. In order to make a more accurate prediction and better understanding of business relationship or other relationships, the research on the position change is needed in the field.

The findings of this thesis have important implications for academia and industry. The academic research on innovation, such as Grewal et al. (2006) or Kim et al. (2011), should also consider the evolution of the network and the changing position of node in evolving network. If a node position can change in the network, the static network property that helps demonstrate innovation performance should no longer be considered the only factor. From a managerial perspective, the software services also follow a life cycle, coherently with prior research on innovation diffusion and a life cycle of innovation adoption (Bass, 1969; Rogers, 2003). Prior research on information system network has missed the change depending on time. In addition, prior research on network analysis has described only the network structure and the evolution rule of a network (Hwang et al., 2009; Valverde and Solé, 2007). Therefore, the findings suggest that analysis method using in this thesis should be applied not only to the analysis of innovation trend in the service network, but also to innovation study more generally.

Overall, the network analysis dealt with in this thesis opens up the possibility for further studies. In order to generalize the findings, they need to be validated with a statistical method for all nodes in the software service network. Finally, the integration of this method into the decision support tool would help service owners to better understand the performance of their services.

Bibliography

Albert, R., Jeong, H. & Barabási, A.-L. (1999). Diameter of the World-Wide Web. *Nature*, 401 (6749), 130-131.

Bass, F. M. (1969). A new product growth for model consumer durables. *Management Science*, 15 (5), 215-227.

Bonacich, P. (1987). Power and centrality: A family of measures. *Am. J. of Sociol*, 92, 1170-1182.

Burt, R. S. (1992). *Structural holes: The social structure on competition*. Harvard Univ. Press, MA.

Campbell-Kelly, M. (2009). Historical reflections: The rise, fall, and resurrection of software as a service. *Communications of the ACM*, 52 (5), 28-30.

Chesbrough, H. W. (2003). *Open innovation: The new imperative for creating and profiting from technology*. Harvard Business School Press, MA.

Chesbrough, H. W. (2011). The case for open services innovation: The commodity trap. *California Management Review*, 53 (3), 5-20.

Everard, A. & Henry, R. (2002). A social network analysis of interlocked directorates in electronic commerce firms. *Electronic Commerce Research and Applications*, 1 (2), 225-234.

Freeman, L. C. (1979). Centrality in social networks: Conceptual clarification. *Social Networks*, 1, 215-239.

Girvan, M. & Newman, M. E. J. (2002). Community structures in social and biological networks. *In Proceedings of the National Academy of Science of the USA*, 99 (12), 7821-7826.

Gloor, P. A., Krauss, J., Nann, S., Fischbach, K. & Schoder, D. (2009). Web science 2.0: Identifying trends through semantic social network analysis. *In Proceedings of the 12th CSE, Vancouver, Canada.*

Granovetter, M. S. (1973). The strength of weak ties. *American Journal of Sociology*, 78 (6), 1360-1380.

Grewal, R., Lilien, G. L. & Mallapragada, G. (2006). Location, location, location: How network embeddedness affects project success in open source systems. *Management Science*, 52 (7), 1043-1056.

Haines, M. N. & Rothenberger, M. A. (2010). How a service-oriented architecture may change the software development process. *Communications of the ACM*, 53 (8), 135-140.

Hansen, M. T. (1999). The search-transfer problem: The role of weak ties in sharing knowledge across organization subunits. *Administration Science Quarterly*, 44 (1), 82-111.

Hargadon, A. B. (2002). Brokering knowledge: Linking, learning and innovation. *Research in Organizational Behavior*, 24, 41-85.

Hwang, J., Altmann, J. & Kim, K. (2009). The structural evolution of the Web2.0 service network. *Online Information Review*, 33 (6), 1040-1067.

Jin, J. H., Park, S. C. & Pyon, C. U. (2011). Finding research trend of convergence technology based on Korean R&D network. *Expert Syst. Appl.*, 38 (12), 15159-15171.

Jovanovic, B. & MacDonald, G. M. (1994). The life cycle of a competitive industry. *J. Polit. Econ.*, 102 (2), 322-347.

Kim, K., Hwang, J. & Altmann, J. (2011). Does Web2.0 foster innovation? An analysis of the openness of the Web2.0 service network. *In Proceedings of the 44th HICSS, Kauai, HI.*

Krackhardt, D. & Stern, R. N. (1980). Informal networks and organizational crises: An experimental simulation. *Social Psychology Quarterly*, 51 (2), 123-140.

Lemmens, C. E. A. V. (2004). *Innovation in Technology Alliance Networks*. Edward Elgar, MA.

Maglio, P. P., Srinivasan, S., Kreulen, J. T. & Spohrer, J. (2006). Service systems, Service scientists, SSME, and Innovation. *Communications of the ACM*, 49 (7), 81-85.

Newman, M. E. J. (2001). Clustering and preferential attachment in growing networks. *Physical Review E*, 64 (2), 025102.

Ogrinz, M. (2009). *Mashup patterns: designs and examples for the modern enterprise*. Addison-Wesley.

Opsahl, T., Agneessens, F. & Skvoretz, J. (2010). Node centrality in weighted networks: Generalizing degree and shortest paths. *Social Networks* 32 (3), 245-251.

O'Reilly (2007). What is Web2.0: design patterns and business models for the next generation of software. *Communications and Strategies*, 65, 17-37.

Papazoglou, M. P. & Georgakopoulos, D. (2003). Introduction: Service-oriented computing. *Communications of the ACM*, 46 (10), 24-28.

Rogers, E. (2003). *Diffusion of innovation*, 5th *ed*. Free Press, NY.

Sasidharan, S., Santhanam, R., Brass, D. J. & Sambamurthy, V. (2011). The effects of social network structure on enterprise systems success: A longitudinal multilevel analysis. *Information Systems Research*, 23 (3-1), 658-678.

Scott, J. (1991). *Social network analysis: A Handbook*. Sage Publication, UK.

Tsai, W. (2001). Knowledge transfer in intraorganizational networks: Effects of network position and absorptive capacity on business unit innovation and performance. *Academy of Management Journal*, 44 (5), 996-1004.

Valverde, S. & Solé, R. V. (2007). Self-organization versus hierarchy in open source social networks. *Physical Review E*, 76 (4), 046118.

Wagner, C. S., & Leydesdorff, L. (2005). Network structure, self-organization, and the growth of international collaboration in science. *Research Policy*, 34 (10), 1608-1618.

Abstract (Korean)

서비스 네트워크는 개방형 혁신 시스템으로 간주된다. 소프트웨어 공급자들이 소프트웨어를 서비스로 제공하고 사용자들이 개방형 인터페이스를 통해 서비스의 기능에 접근하는 동안, 혁신 스타일은 국부적인 혁신에서 집단적인 혁신으로 바뀌고 있다. 이러한 새로운 혁신 트렌드는 소프트웨어 서비스 네트워크 내에서 구조적, 진화적 패턴에 대한 연구들이 진행되도록 이끌어왔다. 그러나 네트워크 구조와 노드 포지션의 정적인 특성에만 관심을 기울였던 이전의 연구들은 진화하는 네트워크의 동적인 특성을 고려하지 못하고 있다. 따라서 본 연구에서는 www.programmableweb.com에 공개된 실증적 데이터를 이용하여 대표적인 소프트웨어 서비스들의 중심성을 분석함으로써 시간에 따른 네트워크 포지션의 변화를 연구한다. 연구 결과에서 각각의 서비스는 전형적인 라이프 사이클을 따르며, 혁신 트렌드는 네트워크 구조의 변화와 더불어 사진 서비스에서 소셜 네트워킹 서비스로 이동하고 있다는 사실을 보여준다. 본 연구는 혁신 연구가 정적인 네트워크 위상과 포지션뿐만 아니라, 진화하는 네트워크에서의 포지션 변화도 고려해야 한다는 것을 시사한다.

주요어 : 서비스 네트워크, Software-as-a-Service, 혁신 트렌드, 중심성, 사회연결망 분석, 복합 서비스, 개방형 혁신.
학 번 : 2012-21042