



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Sistemas

**Análisis basado en grafos para programas de estudio
en computación de universidades del Perú**

TESIS

Para optar el Título Profesional de Ingeniero de Sistemas

AUTOR

Jorge Luis ALVARADO REVATA

ASESOR

John Ledgard TRUJILLO TREJO

Lima, Perú

2024



Reconocimiento - No Comercial - Compartir Igual - Sin restricciones adicionales

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Usted puede distribuir, remezclar, retocar, y crear a partir del documento original de modo no comercial, siempre y cuando se dé crédito al autor del documento y se licencien las nuevas creaciones bajo las mismas condiciones. No se permite aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros a hacer cualquier cosa que permita esta licencia.

Referencia bibliográfica

Alvarado, J. (2024). *Análisis basado en grafos para programas de estudio en computación de universidades del Perú*. [Tesis de pregrado, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e Informática, Escuela Profesional de Ingeniería de Sistemas]. Repositorio institucional Cybertesis UNMSM.

**Metadatos complementarios autor/
asesor**

Datos de autor	
Nombres y apellidos	Jorge Alvarado Revata
Tipo de documento de identidad	DNI
Número de documento de identidad	10279281
URL de ORCID	https://orcid.org/0000-0002-7298-3419
Datos de asesor	
Nombres y apellidos	John Ledgard Trujillo Trejo
Tipo de documento de identidad	DNI
Número de documento de identidad	06187585
URL de ORCID	https://orcid.org/0000-0002-0563-4809
Datos del jurado	
Presidente del jurado	
Nombres y apellidos	Augusto Parcemon Cortez Vásquez
Tipo de documento	DNI
Número de documento de identidad	08634618
Miembro del jurado 1	
Nombres y apellidos	Johnny Robert Avendaño Quiroz
Tipo de documento	DNI
Número de documento de identidad	09739286
Datos de investigación	

Línea de investigación	No aplica.
Grupo de investigación	No aplica.
Agencia de financiamiento	Sin financiamiento.
Ubicación geográfica de la investigación	País: Perú Departamento: Lima Provincia: Lima Distrito: Lima Cercado Jr. Carlos Amezaga No. 375 Universidad Nacional Mayor de San Marcos Latitud: -12.0564232 Longitud: -77.0843327
Año o rango de años en que se realizó la investigación	2022-2024
URL de disciplinas OCDE	2.02.04 -- Ingeniería de sistemas y Comunicaciones https://purl.org/pe-repo/ocde/ford#2.02.04



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERIA DE SISTEMAS E INFORMATICA
Escuela Profesional de Ingeniería de Sistemas

Acta Virtual de Sustentación de Tesis

Siendo las 15:00 horas del día 11 de junio del año 2024, se reunieron virtualmente los docentes designados como miembros de Jurado de Tesis, presidido por el Mg. Augusto P. Cortez Vásquez, Mg. Johnny R. Avendaño Quiroz (Miembro) y el Lic. John L. Trujillo Trejo (Miembro Asesor), usando la plataforma Meet (<https://meet.google.com/wdy-edqm-wpk>), para la sustentación Virtual de la tesis Intitulada: **“Análisis basado en grafos para programas de estudio en computación de universidades del Perú”**, del Bachiller: **Jorge Luis Alvarado Revata**; para obtener el Título Profesional de Ingeniero de Sistemas.

Acto seguido de la exposición de la Tesis, el Presidente invitó al Bachiller a responder las preguntas formuladas por los Miembros del Jurado.

El Bachiller, en el curso de sus intervenciones demostró pleno dominio del tema, al responder con acierto y fluidez las preguntas formuladas por los señores miembros del Jurado.

Finalmente habiéndose efectuado la calificación correspondiente por los miembros del Jurado, el bachiller obtuvo la nota de 18 (dieciocho)

A continuación, el Presidente del Jurado Mg. Augusto P. Cortez Vásquez, declara al Bachiller **Ingeniero de Sistemas**.

Siendo 16:00 horas, se levantó la sesión.

Mg. Augusto P. Cortez Vásquez
Presidente

Mg. Johnny R. Avendaño Quiroz
Miembro

Lic. John L. Trujillo Trejo
Miembro Asesor



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Vicerrectorado de Investigación y Posgrado



Yo **JOHN LEDGARD TRUJILLO TREJO** en mi condición de **asesor** acreditado con la Resolución Directoral N° 000049-2019-EPISI-FISI/UNMSM de la **tesis**, cuyo título es **ANÁLISIS BASADO EN GRAFOS PARA PROGRAMAS DE ESTUDIO EN COMPUTACIÓN DE UNIVERSIDADES DEL PERÚ**, presentado por el **bachiller JorgeLuis Alvarado Revata** para optar el **TITULO DE INGENIERO DE SISTEMAS**, CERTIFICO que se ha cumplido con lo establecido en la Directiva de Originalidad y de Similitud de Trabajos Académicos, de Investigación y Producción Intelectual. Según la revisión, análisis y evaluación mediante el software de similitud textual, el documento evaluado cuenta con el porcentaje de **12%** de similitud, nivel **PERMITIDO** para continuar con los trámites correspondientes y para su **publicación en el repositorio institucional**.

Se emite el presente certificado en cumplimiento de lo establecido en las normas vigentes, como uno de los requisitos para la obtención del TITULO PROFESIONAL DE INGENIERO DE SISTEMAS.

Firma del Asesor

DNI: 06187585

Nombres y apellidos del asesor: John Ledgard Trujillo Trejo

AGRADECIMIENTOS

Agradezco sinceramente a mi asesor Lic. John Trujillo, por su apoyo, confianza y libertad permitida para el desarrollo de esta investigación. También agradezco a todas las personas con las que en algún momento compartí el interés por conocer la historia y el desarrollo de la informática peruana. Finalmente agradecer a mis jurados Mag. Augusto Cortez Vásquez y al Mag. Jonny Avendaño Quiróz, por sus detalladas sugerencias y observaciones.

DEDICATORIA

Esta tesis está dedicada a mi familia: a mi madre Soledad Revatta Escobar †, a mis hermanos Rosario Alvarado y Javier Alvarado y a mi hija Sofía Valentina Alvarado Zeta. También dedico esta tesis a los estudiantes de informática en cualquiera de sus diversas representaciones, quienes en algún momento de su formación requieren conocer las disciplinas, orígenes e influencias que dan forma a su profesión.

RESUMEN

El presente estudio aborda la diversidad y dispersión de propuestas educativas en la educación universitaria de informática en el Perú, con el objetivo de identificar una métrica que permita encontrar proximidades entre distintos planes de estudio así como también respecto a referentes o guías estandarizadas internacionales; se propone la aplicación de la similaridad de grafos para el respectivo análisis comparativo. Los datos analizados será información pública de planes de estudio de escuelas universitarias, así como referentes internacionales en educación informática. En el desarrollo del proyecto se ha elaborado una herramienta web que permite la elección de los planes y guías que se pretende comparar. Este estudio contribuye en el análisis de mejoras de los diseños curriculares de nuestras instituciones de educación superior.

Palabras clave— planes de estudio, grafos, redes, similaridad, curriculas, algoritmos.

ABSTRACT

This study addresses the diversity and dispersion of educational proposals in computer science university education in Peru, with the aim of identifying a metric that allows finding proximities between different study plans as well as with respect to international standardized references or guides; the application of the similarity of graphs is proposed for the respective comparative analysis. The data analyzed will be public information on study plans of university education schools, as well as international references in computer education. In the development of the project, a web tool has been developed that allows the choice of the plans and guides that are to be compared. This study contributes to the analysis of improvements in the curricular design of our higher education institutions.

Keywords— study plans, graphs, networks, similarity, curricula, algorithms.

TABLA DE CONTENIDO

Agradecimientos	ii
Dedicatoria.....	iii
Resumen.....	iv
Abstract	v
Tabla de Contenido	vi
Índice de figuras.....	viii
Índice de tablas	ix
Capítulo I: Planteamiento del problema	1
1.1 Situación problemática	1
Importancia	1
Novedad	2
Viabilidad.....	2
1.2 Formulación del problema	3
1.3 Justificación de la Investigación.....	3
1.4 Objetivos de la Investigación.....	4
Objetivo general.....	4
Objetivo Específicos	4
Capítulo II: Marco teórico	5
2.1 Antecedentes del problema.....	5
El contexto social y científico del Perú.....	5
La diversidad de nombres de carreras de informática en la región.....	5
Identificar relaciones más allá de la dualidad ciencia o ingeniería	6
2.2 Bases Teóricas Generales.....	8
La evolución de la disciplina informática	8
Primero las computadoras.....	8
Computer + Sciences = Computer Science.....	8
De computer sciences a computer science.....	9
"Computer Science" es informática.....	10
Definición de Informática	11
La adopción de innovaciones	13
El desarrollo de los referentes de informática.....	18
La adopción de la informática en el Perú.....	19
El modo de origen de las innovaciones en educación universi- taria de informática	19
La tensión artificial entre ciencia e ingenierías.....	23
Modelos de la teoría moderna de la informática	24
Evolución de las sub-áreas de la informática	24
2.3 Estado del Arte.....	28
El modelo de competencias y los perfiles ACM/IEEE	28
El proyecto Tuning Latinoamérica para informática.....	30
2.4 Búsqueda de semejanzas entre propuestas internacionales y la adopción regional	32

Las relaciones entre contenidos y áreas basada en grafos	33
Modelo basado en grafos.....	34
2.5 Contribución teórica para el proyecto	35
Similaridad de grafos	35
Concepto de isomorfismo y sub-grafos.....	35
Máximo común sub-grafo	36
Métrica de similaridad entre grafos.....	37
Algoritmo de Máximo común sub-grafo.....	37
Cuantificación de relaciones y nodos.....	39
Capítulo III: Hipotesis y Variables.....	40
3.1 Hipotesis General	40
3.2 Hipotesis Específicas	42
3.3 Identificación de variables	43
3.4 Operacionalización de variables	44
3.5 Matriz de consistencia.....	46
Capítulo IV: Metodología	47
4.1 Tipo de Diseño de Investigación	47
4.2 Unidad de análisis	48
4.3 Población de estudio	48
4.4 Tamaño de la muestra	49
4.5 Selección de la muestra	49
4.6 Técnicas de recolección de datos	50
Capítulo V: Resultados y discusión	51
5.1 Análisis e interpretación de la información.....	51
Proceso de clasificación, registro y codificación de los datos	51
Técnica analítica	58
5.2 Pruebas de hipótesis.....	59
5.3 Presentación de Resultados.....	60
Caso 1 Hipotesis Secundaria 1	60
Caso 2 Hipotesis Secundaria 1	63
Caso 3 Hipotesis Secundaria 2	65
Caso 4 Hipotesis Secundaria 2	67
Caso 5 Hipotesis Secundaria 2	69
Caso 6 Hipotesis Secundaria 1	71
5.4 Resultados	72
5.5 Conclusiones	74
5.6 Trabajos futuros	75
Bibliografía	76
Appendix A: ANEXOS	79
A.1 Documento de codificación de datos	79
A.2 Repositorio de imagenes y datos codificados	79

ÍNDICE DE FIGURAS

<i>Número</i>	<i>Página</i>
5.1 Sis2009 ACM68 MCS modelo 1	61
5.2 Sis2009 ACM68 MCS modelo 2	61
5.3 Sis2017 ACM68 MCS modelo 1	63
5.4 Sis2017 ACM68 MCS modelo 2	63
5.5 Sis2009 Sis2017 MCS modelo 1	65
5.6 Sis2009 Sis2017 MCS modelo 2	65
5.7 Sis2017 y ACM 2013 MCS modelo 1	67
5.8 Sis2017 y ACM 2013 MCS modelo 2	67
5.9 IngInform2020 y ACM 2013 MCS modelo 1	69
5.10 IngInform2020 y ACM 2013 MCS modelo 2	69
5.11 IngSistemas2018 y ACM 68 MCS modelo 1	71
5.12 IngSistemas2018 y ACM 68 MCS modelo 2	71

ÍNDICE DE TABLAS

<i>Número</i>	<i>Página</i>
2.1 Sub-áreas de la informática.	25
2.2 Algoritmos y estructuras de datos	26
2.3 Lenguajes de programación	27
2.4 Sistemas Operativos y Redes.....	27
2.5 Competencias Generales	28
2.6 Competencias Específicas para informática.....	29
2.7 Resultados Intensidad y cobertura de competencias: Ingeniería en Siste- mas ORT.....	29
2.8 Competencias genéricas del informe Tuning	31
2.9 Competencias de la disciplina	32
3.1 Matriz de operacionalización de variables	44
3.2 Matriz de consistencia	46
4.1 Matriz de planes de estudio considerados en el estudio	49
4.2 Matriz de guías internacionales	49
5.1 Guía ACM 68	52
5.2 Guía ACM 2001	53
5.3 Guía ACM 2013	54
5.4 Sistemas 2009 con Acm68	55
5.5 Sistemas 2017	56
5.6 Tabulación de datos sobre comparaciones	59
5.7 Tabulación de datos sobre comparaciones	61
5.8 Comparación G-ACM 68 y H-Sistemas 2009	61
5.9 Tabla de resultados de proporcionalidad.....	62
5.10 Tabla de Hipotesis vs Resultados	62
5.11 Tabulación de datos sobre comparaciones	64
5.12 Comparación G-ACM 68 y H-Sistemas 2009	64
5.13 Tabla de resultados de proporcionalidad.....	64
5.14 Tabla de Hipotesis vs Resultados	64
5.15 Tabulación de datos sobre comparaciones	66

5.16	MCS	66
5.17	Tabla de resultados de proporcionalidad.....	66
5.18	Tabla de Hipotesis vs Resultados	66
5.19	Tabulación de datos sobre comparaciones	67
5.20	MCS	68
5.21	Tabla de resultados de proporcionalidad.....	68
5.22	Tabla de Hipotesis vs Resultados	68
5.23	Tabulación de datos sobre comparaciones	70
5.24	MCS	70
5.25	Tabla de resultados de proporcionalidad.....	70
5.26	Tabla de Hipotesis vs Resultados	70
5.27	Tabulación de datos sobre comparaciones	72
5.28	MCS	72
5.29	Tabla de resultados de proporcionalidad.....	72
5.30	Tabla de Hipotesis vs Resultados	72

PLANTEAMIENTO DEL PROBLEMA

1.1 Situación problemática

El presente estudio se pregunta si es posible encontrar semejanzas en la diversidad de ofertas de programas de estudio de carreras de informática en el país. La adopción de las guías o referentes internacionales no se ha documentado adecuadamente en distintas casas de estudio, además que por muchas razones entre ellas las culturales, la informática peruana se ha desarrollado en distintas vertientes y enfoques.

Para alcanzar nuestro objetivo usaremos la información pública de planes de estudio de programas de pre-grado, así como información pública de los referentes o guías internacionales. El estudio busca identificar relaciones que al margen de las denominaciones y tiempo de aparición de un plan de estudio este tenga cierto nivel de semejanza con otro plan o guía. Los resultados se medirán con variables cuantitativas y con un proceso de codificación que permita obtener resultados.

Este estudio es la continuación de un estudio previo que abordaba los elementos sociales y culturales que moldearon el desarrollo de la informática peruana. En ese estudio previo se identificó que era posible que una institución preserve su tradición o identidad incorporando a nivel de planes de estudio, las mejoras que se proponen en los perfiles internacionales, sin que ello las obligue a crear nuevas escuelas. Este estudio expone un proceso y unas técnicas de análisis a nivel micro usando como apoyo la teoría de redes en grafos.

Importancia

Esta propuesta es importante para contextualizar una enfoque de mejora continua en la educación informática del país, no solo bajo una mirada dual entre ciencia o ingeniería; por el contrario se busca introducir una interpretación más amplia de la naturaleza de las escuelas universitarias de informática; se espera incorporar elementos adicionales para una lectura de la historia y desarrollo de la comunidad

informática peruana con elementos que permitan la adopción de innovaciones que aparecen en otras instituciones rectoras.

Novedad

La novedad de esta propuesta es que incorpora un análisis micro de los planes de estudio en un modelamiento macro de innovación, es decir al problema de como mejorar las propuestas educativas y acercarlas a las guías internacionales, se plantea el modelamiento y la comparación cuantitativa que permita a las instituciones analizar y planificar su estrategia de adopción de innovaciones educativas. De modo que es posible entender el desarrollo de una profesión como la ingeniería de sistemas, la ingeniería informática y su relación con la historia de la profesión informática peruana así como su relación con el desarrollo internacional.

Viabilidad

Este estudio permitirá a los futuros estudiantes, a quienes se encuentran estudiando y a los diseñadores de planes de estudio validar la relación existente entre los planes de estudio y referentes internacionales. No solo por equivalencias de nombres sino por contenidos en cursos como en distintos enfoques de los planes de estudio.

1.2 Formulación del problema

La pregunta principal de la investigación es:

1. P1.(H1) ¿Que relaciones podemos identificar en los planes de estudio de programas de informática con otros planes o guías internacionales?

Las pregunta principal se puede expresar con variantes de la siguiente manera:

1. P2. (H2)¿Que relaciones existe entre los programas de estudio y referentes internacionales en distintos periodos de tiempo?
2. P3. (H3)¿Que relaciones existe entre los planes de estudio en distintos periodos de tiempo o distintas casas de estudio?

El proyecto elabora un prototipo de herramienta software para realizar comparaciones y encontrar indicadores cuantitativos modelando planes de estudios vigentes y ofrecidos actualmente en universidades del país así como referentes internacionales. La idea es que en base a este desarrollo se pueda construir una herramienta de comparación para incorporar distintos planes de estudio. El proyecto espera contribuir con nuevos elementos para un mejor entendimiento entre las comunidades de la informática peruana.

1.3 Justificación de la Investigación

El escenario de la educación en informática en el país tiene distintas interpretaciones. Contribuir con herramientas para mejorar la adopción de referentes internacionales o el reconocimiento de las propuestas con equivalencias internacionales es un elemento importante para la acreditación, difusión e de internacionalización de las propuestas educativas.

Para un estudiante o postulante a una carrera de ingeniería asociada a la informática o los sistemas de información, es importante visibilizar el alcance de los planes de estudio respecto a lo que conocemos sobre la informática. Para una casa de estudio,

el beneficio es la identificación de sus propuestas educativas como parte de una propuestas de educación internacional o al menos con indicios o rasgos que hagan referencia a la evolución de la educación en informática.

1.4 Objetivos de la Investigación

Objetivo general

Adaptar y aplicar conceptos de redes y similaridad de grafos en el análisis de relaciones entre programas de estudios y guías de perfiles profesionales.

Objetivo Específicos

1. Identificar similaridad entre planes de estudio y guías de perfiles.
2. Identificar similaridad entre planes de estudio en distintos periodos o entre casas de estudio.

Capítulo 2

MARCO TEÓRICO

2.1 Antecedentes del problema

El contexto social y científico del Perú

El contexto económico y político que identificamos en la presente investigación como influyentes en el desarrollo de las profesiones universitarias y por ende de la informática lo encontraremos en (McLauchlan de Arregui, 1990) en donde exponen los problemas crónicos del sistema universitario; Gonzales brinda una visión del impacto del terrorismo en la economía del país (Gonzales de Olarte, 1990). Cualquier avance logrado en el periodo de la década de los 70, quedan afectados por las condiciones políticas y económicas del país de la década de los 80.

Yamada encuentra una relación entre la baja calidad universitaria y el sub-empleo, entendido como el ejercicio profesional en labores alejadas de los campos de estudio (Yamada et al., 2014)

Quien mejor resume la realidad de la ciencia en el Perú en la actualidad es (Ortiz, 2020) donde textualmente indica que la ciencia es muy pequeña en el país. Por lo que los talentos que emigran y se forman en escuelas de otras latitudes lamentablemente no tienen incentivos para regresar y renovar el sistema académico universitario. Esta realidad es compartida por muchos países de la región.

La diversidad de nombres de carreras de informática en la región

En el Perú y en parte de la región hispanohablante se presentan características especiales en el desarrollo de las escuelas universitarias de informática. Existe una diversidad de escuelas creadas en facultades de ingeniería, con denominaciones variadas (Ramos et al., 2013). Esas características difieren de algún modo al desarrollo de las profesiones en otros sistemas educativos de países más desarrollados.

La característica más visible es la diferencia de nombres de estas profesiones. Pero la mayor diferencia no es precisamente la denominación, sino su orientación a instrumentalizar la informática para orientar la formación en aspectos pragmáticos.

Aún con ese panorama, los programas de estudio se han ido adaptando a los tiempos y también han aparecido nuevas propuestas educativas. Los referentes internacionales bajo el dominio de ACM/IEEE centra su difusión respecto al concepto de "Computer Science" como un campo de amplia diversidad que se expresa en distintos perfiles profesionales que no se consideran profesiones de ingeniería (a excepción de ingeniería del software). Eso ha generado debates dentro del seno de la comunidad académica peruana, estableciéndose en algunos casos una tensión artificial entre los enfoques educativos.

Una manera de abordar los distintos enfoques interpretativos del desarrollo de la informática peruana se aborda en un informe del Colegio de Ingenieros del Perú (2006) (CIP, 2006) en el que se recomendaba que las ingeniería de sistemas se orienten al modelo internacional de ingeniería de sistemas, (INCOSE, 2011); en la práctica, las escuelas de ingeniería de sistemas se han mantenido como tal y continúan adaptándose bajo sus propias políticas educativas, siendo considerada a nivel laboral como una profesión más de la esfera informática.

Identificar relaciones más allá de la dualidad ciencia o ingeniería

Varios autores han intentado estudiar el fenómeno del desarrollo de la informática peruana y de la región, como profesión universitaria; (Álvarez, et, al., 2008) nos ofrece unas alternativas de referencias internacionales que considere a las profesiones pre-existentes (Alvarez et al., 2008); (Rivera et al., 2008), nos menciona la conexión histórica de la escuela de computación en San Marcos con la ingeniería de sistemas de la misma casa de estudios; (Sabin, et, al., 2016) (Sabin et al., 2016) indica la orientación a Information Technology para las profesiones de pre-grado en la región.

El trabajo más realista respecto a no intentar forzar un cambio de denominación a las escuelas existentes de las carreras de sistemas, informática y computación y por el contrario diseñar una especie de filtro nuevo que permita realizar equivalencias

entre perfiles locales versus perfiles globales, se planteó en el trabajo de (Ramos, et, al., 2013)(Ramos et al., 2013).

2.2 Bases Teóricas Generales

La evolución de la disciplina informática

Primero las computadoras

Para entender que es informática “computer science” debemos primero averiguar como llegamos a ese nombre tan interesante y provocador. El cuerpo de conocimientos que comienzan a dar forma a lo que conocemos como “computer science” se inicia por los años 40 del siglo pasado, con la intercepción de la teoría de algoritmos, la invención del programa almacenado, los trabajos de Alan Turing, Kurt Gödel, Alonzo Church, el desarrollo del cálculo lambda, los problemas computables y no computables y el desarrollo de las primeras computadoras análogas y posteriormente digitales, con las innovaciones de Vannevar Bush, Aiken, Zuse, Von Newmann, entre muchos mas. Todos esos conocimientos confluyen y para la década de los 50, comienzan a denominarse a todas estas áreas de estudio “computer sciences” es decir ciencias de la computadora.

Como sucedió con la invención de otras ciencias modernas (como la administración, la ingeniería industrial, etc), no sucede espontáneamente; por el contrario es el resultado de múltiples fuerzas que confluyen en este campo de estudio, también de un juicioso análisis del momento visionado por personas que deciden apostar por el naciente campo de estudio.

En la actualidad “Computer science” se intercambia con “computing science” (ciencia de la computación) o “computing (computación), un uso reciente en español es el de ciencia de la computación; desde su creación hasta la actualidad el nombre representa muchos conceptos que se superponen y se traslapan para dar forma a una especie de paraguas que agrupa varios sub-campos; todos vinculados con la automatización y digitalización de la información.

Computer + Sciences = Computer Science

Como tal el origen del termino “computer science” en el habla inglesa comienza a establecerse como el nombre de una nueva ciencia (entendida como un grupo de

conocimientos y prácticas) no una ciencia dura y escrita en piedra, sino un conjunto de campos de conocimiento alrededor de la computadora. La aparición de la computadora digital, fue un salto cuantitativo en el trabajo matemático que hasta ese tiempo era llevado por “computer” (personas) que realizaban los cálculos más tediosos y operativos, de manera manual; la computadora digital transforma esos trabajos y se convierte en el primer objeto de estudio de este nuevo cuerpo de conocimientos.

En el año 1953 en la universidad de Cambridge se crea el primer diploma en “computer science” en el Reino Unido; en los Estados Unidos, Lous Fein hace en un informe (1957) (Fein, 1959) para la universidad de Stanford, donde plantea la urgente necesidad de integrar los conocimientos de educación en los siguientes temas: desarrollo de las computadoras, procesamiento de datos, investigación operativa, programación lineal, psicología neuronal, teoría de juegos, inteligencia artificial, análisis numérico, entre otros campos; en el informe llama a esta confluencia: “Computer sciences” .

Otro papel importante en este desarrollo es el que juega la industria de las computadoras digitales de ese momento, IBM con su enorme necesidad de personal, facilitó la entrega en los campus de los modelos más recientes de sus computadoras, de ese modo se estableció la semilla de innumerables y nuevos usos de la computadora en las universidades. Algunas universidades habían sido partícipes de la creación de los modelos primarios de la computación digital, otras sin embargo encontrarían en esta invitación de la industria una manera de llevar a la práctica muchos problemas de investigación de ese tiempo. Con ello el contar con una computadora no establecía un nuevo campo de estudio necesariamente por lo que haría falta otras acciones que permitirían el auge de “computer science”.

De computer sciences a computer science

Para inicios de los 60, empezaba a establecerse en la comunidad industrial, académica y en el lenguaje, el nombre de “computer sciences” para describir las diversas áreas de conocimientos vinculados con las computadoras, pero es gracias a la influencia de algunos personajes del mundo académico que el campo emerge en los Estados Unidos, uno de esos personajes es George Forsythe (Knuth, 1972) un destacado matemático especializado en métodos numéricos, quien realiza un trabajo de hormiga en

convencer, convocar y crear una organización en el departamento de matemáticas de la universidad de Stanford, para el estudio de la “computer science” , su trabajo no se limitó a la universidad Stanford por el contrario fue un difusor y comunicador con el objetivo de establecer el campo como un nuevo horizonte a ser estudiado y desarrollado; otras figuras que destacan en su tiempo son Alan Perlis, el propio Louis Fein como hemos indicado antes, Richard Hamming, entre otras figuras.

Es importante indicar que para ese tiempo ya se establecían distintos programas de estudio del campo pero con distintos nombres, el trabajo de los precursores como George Forsythe fue darle un horizonte claro a los objetivos de la confluencia y con ello se pierde la “s” y queda en el tiempo como “computer science” .

Un concepto tan amplio y disperso requiere ponerle ciertos límites, cabe indicar que el termino “computer science” (ciencia de la computación) en la actualidad, es un nombre de varias cosas, el nombre de una entidad con muchos rostros y formas. Esa diversidad y amplitud es fuente de controversia hasta nuestros días, “computer science” o “computing science” o “computing” para denominar el campo en lengua inglesa. Para el practicante o el que ejerce el campo, se tienen palabras como “computer scientist” o “computing scientist” . La traducción literal del termino trae consigo algunos problemas para su empleabilidad según el contexto cultural y social del país.

“Computer Science” es informática

En español tenemos una palabra para representar todo lo que puede parecer confuso en ingles, el termino es “informática” . El origen de esta palabra tiene raíces en la escuela europea de la post-guerra. Palabra que se establece en la escuela Alemana y Europea de esos años, Karl W. Steinbuch acuña el termino informatik en 1957 y Philippe Dreyfus, difunde el termino “informatique” , el cual lo establece en 1962, para denominar al conocimiento dedicado al procesamiento automático de información, está información se encuentra documentada por un libro editado por Friedrich L. Bauer (Bauer, 2010); este termino es incluso aceptado por la academia de la lengua francesa. En ingles también existe dicha palabra: “informatic” , pero en el año 1962, Walter F. Bauer (un pionero en el uso de las computadoras electrónicas en los Estados Unidos), crea una empresa de software y una marca comercial con ese termino,

razón por la que la palabra “informatic” empezó a desarrollarse más como un nombre comercial que por su significado, de igual manera la frase computer science empezaba a usarse en el idioma inglés para expresar la nueva disciplina.

El término informática cumple muy bien el papel de ser un nombre que concentra los significados y usos de este cuerpo de conocimiento, incluso resulta manejable y permite describir el trabajo de una persona que se dedica al campo, aunque igual su generalidad no describe mucho lo que llegue a realizar una persona. Así que podría decirse que el término tampoco se usa demasiado de esa forma, por el contrario los usos del lenguaje se centran principalmente en los perfiles tecnológicos de los informáticos.

Definición de Informática

Se puede entender a la informática describiendo algunas de sus características como el desarrollo, aplicación y estudio de algoritmos y su aplicación en otros ámbitos. Un algoritmo es una entidad compuesta de instrucciones finitas (palabras) que en conjunto resuelven un problema, obviamente no es un grupo de palabras al azar, cada palabra es un simbolismo que permite manipular el flujo de la información y usar la memoria y capacidad de una computadora. Cuando esos algoritmos se escriben ya en un lenguaje artificial, que la computadora entiende, se convierte en lo que conocemos como programación; la programación es una disciplina también por sí misma y un sub-campo si queremos organizar a la informática, aunque su carácter pragmático significa que quien programe no necesariamente se dedica a desarrollar la informática, en realidad alguien podría inventarse algoritmos sin necesidad de lenguajes de programación, y alguien podría crear un éxito comercial programando (como ha pasado innumerables veces) sin necesariamente ser un experto en informática.

Los problemas resueltos con algoritmos se conocen como problemas computables, así también existen problemas que no son computables (es decir que no existe un algoritmo eficiente para encontrar una solución) de manera que la informática también estudia esos problemas y encuentra formas de acercarse a la solución. Una herramienta del pensamiento usada por la informática es la capacidad de abstracción que tiene la mente humana para simplificar, reducir complejidad y modelar situaciones con

palabras, dibujos, formas que a su vez permiten seguir construyendo nuevas formas; de modo que con esa organización de conceptos logramos una explicación y solución de problemas. El modelamiento, llamado abstracción en informática, no es propia del campo pero sirve como instrumento para resolver problemas. Lo interesante es que en muchos casos ese modelamiento se hace siguiendo lógica formal y se traduce en palabras que las mismas computadoras son capaces de entender, por lo que la capacidad de almacenar información, procesar un lenguaje y resolver problemas con esos lenguajes son cosas que se pueden hacer en la informática.

Una definición que aparece en una conferencia de George Forsythe (Forsythe, 1968) y que se mantiene vigente hasta la actualidad; la define como una pregunta: ¿Que puede ser automatizado? Tiene esa pregunta implicancias importantes hasta nuestros días y cada cierto tiempo recibe distintos nombres, acaso no está esa pregunta en el fondo de palabras comerciales como "Transformación digital" , "Agilidad" , "digitalización" , "aprendizaje automático" ; incluso la sobre-automatización del trabajo tiene implicaciones sociales importantes sobre el futuro del trabajo del ser humano. Como tal es importante tener una definición lo suficiente amplia pero clara de lo que es la informática.

La definición de la enciclopedia Británica(Britannica, 2020) la define como : "Informática es el estudio de las computadoras, de la computación, incluye aspectos teóricos y fundamentos algorítmicos, hardware y software, y los usos para el procesamiento de información" . Peter J. Denning, define el cuerpo de conocimientos como: "el estudio sistemático de procesos algoritmos que describen y transforman información: Su teoría, análisis, diseño, eficiencia, implementación y aplicación. ¿Que puede ser automatizado (de modo eficiente)?" .

Estas características sin embargo dejan de lado el impacto que ha tenido la informática en otras disciplinas como en la sociedad, el contexto y las personas que trabajan en ella. En palabras de Denning: Denning (2003, p.405)

Esta caracterización común es demasiado austera. Solo insinúa parte de la riqueza total de la disciplina. No llama la atención sobre las conexiones entre el conocimiento informático y las preocupaciones de las personas a las que contribuye este conocimiento, en particular las preocupaciones

universales por la fiabilidad, confiabilidad, solidez, integridad, seguridad y modificabilidad de los sistemas informáticos. Oculta el contexto social e histórico del campo y los valores de las personas que lo practican.

Por lo tanto es reconocible que la informática es más que el concepto central de ciencia que muchas veces se le otorga y que el objeto de estudio se desplazará desde su aplicabilidad a su base teórica o de las relaciones que pueda establecerse con otros campos, así como de las implicaciones sociales que conlleva su práctica.

La adopción de innovaciones

En esta sección se desarrollan los conceptos de adopción de innovaciones para poder entender que factores han influido y configurado nuestro sistema educativo en informática.

En la siguiente sección, tomaremos conceptos del libro *Difusión de Innovaciones* de Everett. M. Rogers:

(a) Difusión: Rogers et al. (1962, p.35):

La difusión es el proceso por el cual una innovación es comunicada a través de ciertos canales en el tiempo entre los miembros de un sistema social. La difusión es un tipo especial de comunicación relacionado con la expansión de los mensajes que se perciben como ideas nuevas. La comunicación es un proceso en el que los participantes crean y comparten información entre sí con el fin de llegar a un entendimiento mutuo. La difusión tiene un carácter especial debido a la novedad de la idea en el contenido del mensaje. Por lo tanto cierto grado de incertidumbre y riesgo percibido está involucrado en el proceso de difusión. Un individuo puede reducir este grado de incertidumbre mediante la obtención de información. La información es una diferencia en un tipo de energía que afecta a la incertidumbre en una situación donde existe una elección entre un conjunto de alternativas. Los principales elementos de la difusión de

nuevas ideas son: (1) una innovación (2) que se comunica a través de ciertos canales (3) en el tiempo (4) entre los miembros de un sistema social.

- (b) Innovación: Everett. M. Rogers define a una innovación: como una idea, práctica, u objeto que es percibido como nuevo por un individuo u otra unidad de adopción. Si una idea parece nueva para un individuo, es una innovación. La novedad en una innovación no tiene necesariamente por qué implicar nuevo conocimiento.

Rogers describe que no todas las innovaciones son deseables, ni todos los posibles adoptantes de la innovación pueden desear la innovación si sus condiciones difieren. Algunas innovaciones pueden ser dañinas y anti-económicas.

- (c) Atributos de la innovación:

Rogers describe los siguientes atributos de una innovación:

Ventaja relativa: Es el grado en que una innovación es percibida como mejor a ideas existentes. Este grado puede ser medido en términos económicos, factores de prestigio social, conveniencia y satisfacción. No importa cuán objetiva sea la ventaja.

Compatibilidad: Es el grado en que una innovación es percibida como consistente con los valores existentes, experiencias pasadas, y con las necesidades de los potenciales adoptantes.

Complejidad: Es el grado en que una innovación es percibida como difícil de comprender y usar.

Capacidad de prueba (Trialability): Es el grado en que una innovación puede ser probada de forma básica. Una innovación que se puede probar representa menos incertidumbre para el individuo que está considerando la innovación.

Observabilidad: Es el grado en que los resultados de una innovación se hacen visibles a otros. La visibilidad estimula la discusión entre pares de la nueva idea, así como amigos o vecinos de un adoptante, solicitando información sobre la innovación.

Las innovaciones, percibidas por las personas, que tienen mayor ventaja relativa, compatibilidad, capacidad de prueba, observabilidad, y menos complejidad se adoptarán con mayor rapidez que otras innovaciones.

- (d) La re-inención: Rogers define la re-inención como el grado en que una innovación es cambiada o modificada por un usuario o unidad, en el proceso de adopción o implementación.
- (e) Canales de comunicación: Rogers describe el proceso de difusión como el intercambio de información en donde un individuo o unidad de adopción comunica una nueva idea a uno o varios otros individuos o unidades de adopción a través de un canal de comunicación. El canal es el medio por el cual se realiza el intercambio, puede ser los de tipo tradicional como radio, televisión, periódicos, como también canales de comunicación interpersonales, que incluyen la comunicación cara a cara. La comunicación interactiva vía Internet también es un canal más a considerar.
- (f) Afinidad y diferencias de los individuos y unidades en la difusión: Rogers explica que una difusión se expande si existe cierta afinidad entre los elementos de la comunicación y tiene más problemas si es que existe demasiadas diferencias en las características de los individuos o unidades. El autor define la homofilia como la cercanía en características de los comunicantes tales como: creencias, educación, estatus socioeconómico y gustos. La heterofilia por el contrario se define como el grado de diferencia en estas características entre los individuos o unidades. Es necesario indicar que un cierto grado de heterofilia es necesario en un grupo de afinidad. La característica de heterofilia se presenta respecto a la innovación. Caso contrario en un estado de intercambio con bastante similitud, no se presenta innovación.
- (g) Tiempo: El tiempo en el marco conceptual de la teoría de difusión, se establece como el tercer elemento en el proceso de difusión. El tiempo se identifica en el proceso de decisión de innovación (1), en el momento en que se adopta o rechaza. La capacidad de innovación (2) o adopción (temprana o tardía) de un individuo o unidad que se encuentra adoptando una innovación, respecto a otros miembros de un sistema. Por último, el ratio de adopción en un sistema

(3), medido como el número de miembros de un sistema, que adopta una innovación en un determinado periodo de tiempo.

- (h) El proceso de decisión de la innovación. Rogers define el proceso de decisión como el proceso por el cual un individuo o una unidad pasa de conocer una innovación, a la formación de actitudes acerca de esta, luego a la decisión de adoptarla o rechazarla, luego a implementarla y usarla y finalmente a la confirmación de esta decisión. Estos pasos se conceptualizan de la siguiente manera: (1) Conocimiento, (2) persuasión, (3) decisión, (4) implementación, y (5) confirmación.

- (1) El Conocimiento, es obtenido cuando un individuo o unidad aprende de la existencia de la innovación y gana algún entendimiento de cómo funciona.
- (2) Persuasión, toma lugar cuando un individuo se crea una favorable o negativa actitud hacia la innovación.
- (3) La decisión, ocurre cuando un individuo se involucra en actividades que lo llevan a adoptar o rechazar una innovación.
- (4) La implementación, significa que un individuo usa la innovación. En la implementación es en donde puede ocurrir la re-innovación.
- (5) La confirmación, se produce cuando un individuo busca el fortalecimiento de una decisión-innovación que ya se ha hecho, pero él o ella puede revertir esta anterior decisión si están expuestos a mensajes contradictorios acerca de la innovación.

El proceso de decisión de innovación es una actividad de búsqueda y procesamiento de información en el cual el individuo obtiene información para gradualmente reducir la incertidumbre de la innovación. Normalmente ocurre en ese orden, a menos que existan escenarios de autoridad en donde la adopción es impuesta.

El periodo de decisión de la innovación es la amplitud de tiempo requerido en el proceso de decisión de la innovación. Algunos adoptan rápidamente otros se toman mucho más tiempo.

Según la teoría, este proceso descrito funciona mejor para individuos, para unidades de adopción como organizaciones existen otras complicaciones. En las organizaciones, al existir objetivos y funciones establecidas, la innovación puede requerir la adaptación de la organización.

- (i) La capacidad de innovación y las categorías de adoptantes. Rogers define a la capacidad de innovación como el grado en el cual un individuo o unidad de adopción asume relativamente pronto una innovación respecto a otros miembros del sistema. Las categorías son las siguientes: (1) innovadores, (2) adoptantes tempranos, (3) mayoría temprana, (4) mayoría tardía y (5) rezagada. La tasa de adopción. Se define como la velocidad en que una innovación es adoptada por los miembros del sistema social. Es la medida de adopción en el sistema en lugar de la adopción por el individuo o unidad.
- (j) Sistema Social. Rogers define al sistema social como un conjunto de unidades interrelacionadas que están involucradas para la resolución conjunta de problemas para lograr un objetivo común. Los miembros pueden ser individuos, grupos informales, organizaciones, y/o otros sub-sistemas. La difusión ocurre en el sistema, la estructura social del sistema, afecta de diferentes maneras la difusión. El sistema social constituye el espacio en el cual la innovación se difunde. Los elementos que pueden afectar la difusión en el sistema son: Normas, los líderes de opinión y agentes de cambio, tipos de decisiones de innovación y las consecuencias de la innovación. La capacidad de innovación de los individuos o unidades depende de las características del individuo como del sistema social del que es miembro.
- (k) Líderes de opinión y agentes de cambio. Rogers expone que el miembro más innovador de un sistema, es muchas veces percibido de modo fuera de lo común en el sistema social. Puede que tenga una baja credibilidad respecto al promedio del sistema, su rol en la difusión puede ser limitada.

Los líderes de opinión en cambio pueden influir en otros miembros del sistema. Este papel no es una posición formal o estatus en el sistema. Básicamente existe por sus capacidades, su accesibilidad social, y conformidad con las normas del sistema. Cuando el sistema se orienta al cambio, el líder puede reflejar

innovación, pero cuando las normas del sistema se resisten al cambio, el comportamiento del líder refleja estas normas. Muchos sistemas tienen líderes innovadores y aquellos que se oponen. Los agentes de cambio, vinculados con una organización o negocio que brinda un servicio o producto (agencia) intentan influir en la decisión de innovación en la dirección que desea la agencia. Los agentes de cambio pueden utilizar a los líderes de opinión en actividades de difusión. Se establecen tres tipos de decisiones de innovación: a) Opcional, b) Colectiva y c) Autoritaria.

- (l) Consecuencias de la innovación. Rogers define a las consecuencias de una innovación de la siguiente manera: 1) Deseable versus indeseable, 2) Directa versus indirecta y 3) previstas versus imprevistas.

Estos conceptos son importantes para este estudio pues permiten poner en contexto, las posibles causas y efectos del sistema académico creado en el país desde la llegada de las primeras computadoras. Así también se considera que contribuye a establecer posibles estrategias y mecanismos de apoyo para la adopción de innovaciones académicas en informática en las instituciones.

En lo que respecta al estudio consideramos tres tipos de innovaciones:

- 1) Innovación de área de conocimiento
- 2) Innovación de competencias profesionales
- 3) Innovación de perfil profesional

En lo que respecta a innovación de perfil profesional principalmente se hará referencia a la creación de nuevos programas educativos que aborden un conjunto de competencias y nuevas áreas de especialidad.

El desarrollo de los referentes de informática

En lo que respecta al desarrollo de los perfiles y áreas de conocimiento de la computación consideraremos a Perekatov (Perekatov, 2008), quién explica el desarrollo de los perfiles académicos de computación en los Estados Unidos, desde una perspectiva ajena al análisis de los propios académicos norteamericanos; consideraremos también algunos elementos de los modelos curriculares que aparecen en cada periodo de

desarrollo de las guías curriculares: Modelo-68 (Atchison et al., 1968), de la década de los 90 (Tucker, 1991), (acm, 1991), y los últimos perfiles y áreas del 2001 (The Joint Task Force on Computing Curricula, 2001), 2005 (Shackelford et al., 2006), 2013 (Joint Task Force on Computing Curricula & Society, 2013). Observaremos la evolución de las áreas y perfiles profesionales de la computación internacional y su influencia en las escuelas peruanas. Un informe de IFIP/UNESCO (Mulder & van Weert, 2001) propone un marco de trabajo para generar curricula adaptadas a las capacidades de cada país, en especial a los países en vías de desarrollo.

La adopción de la informática en el Perú

En el Perú las primeras computadoras digitales llegaron como herramientas para nuevas aplicaciones empresariales y de gobierno, así como educativas; la educación en las tecnologías de informática dada la apertura del mercado se centraron en la aplicabilidad. Solo en puntos neurálgicos se desarrolla una comunidad que empieza a adoptar y dar forma a una academia orientada al desarrollo del conocimiento de la informática para formar cuadros universitarios.

A nivel universitario, la informática se instrumentalizó como una profesión de ingeniería. En la mayoría de casos se posicionan en el mercado como profesiones tecnológicas; para entender ese estado actual es preciso incorporar algunos elementos que explican como llegamos a esa situación.

El modo de origen de las innovaciones en educación universitaria de informática

En los siguientes puntos revisaremos elementos que condicionan el origen de la educación informática peruana. Mencionaremos el desarrollo de propuestas educativas en instituciones con mayor trayectoria.

(a) El origen de la computación en San Marcos.

El documento que hace referencia al origen de las profesiones informáticas en el Perú es el informe del colegio de ingenieros(a) del Perú CIP (CIP, 2006). Es el primer documento de alcance público que aborda las diferencias de denominaciones y el origen de las profesiones informáticas.

En el documento se menciona el convenio de la universidad Nacional de San Marcos con la universidad de Grenoble(Francia) de 1969, lo que daría inicio al programa de licenciatura en computación. En el informe, los autores realizan una observación que en primera instancia podría requerir una interpretación adicional.

CIP (2006, p.84)

El enfoque del programa Académico de Computación era eminentemente científicista, con muchos cursos de matemáticas..., luego venían los cursos de estadística, probabilidades..., luego los de la carrera...Pero no había nada análisis y diseño de software ni cursos de gestión de tecnologías de la información, tampoco nada de economía, administración o contabilidad...

Los autores de esa sección, asocian conceptos como: software y tecnologías de la información para re-afirmar el carácter científico de la propuesta de computación del 69, pero no debemos olvidar el contexto en el que se crea esa escuela; con el apoyo de la academia francesa, que precisamente en ese tiempo es que define el concepto de informática como equivalente a ciencia de la computación. El documento ACM68 tiene cierta semejanza con lo que se enseñaba en la escuela de computación de San Marcos. San Marcos es la primera universidad que tiene un programa de estudios en informática con la denominación de computación desde los 70 hasta el año 1997 en que se transformaría en ingeniería de sistemas.

La explicación de esa transformación corresponde a una respuesta a la difusión del concepto de ingeniería de sistemas, fuertemente difundido en esa década, Rivera (Rivera et al., 2008) ofrece una aproximación en donde expone que las necesidades del mercado y la política del rector del momento que impusieron la decisión de crear una escuela de ese tipo. Se puede agregar a su análisis, las siguientes consideraciones: a) La política adoptada por el rector de ese entonces (fuerza política), b) las necesidades de independencia de la escuela (fuerzas políticas internas), c) el dominio de la ingeniería de sistemas como referente universitario (difusión extendida de un referente profesional); d) la

poca difusión de la profesión informática como ciencia, a esto se suma e) el entorno negativo para la investigación que empezaría a gestarse desde los 80 y 90 en la sociedad.

Esos elementos extinguirían el programa de licenciatura y definirían la creación de un programa utilizando el término de ingeniería de sistemas. Sin embargo, parte del colectivo humano que representaba a esta escuela educativa de computación (alumnos, profesores, trabajadores), continuó siendo parte del desarrollo de la nueva facultad. Por lo que puede afirmarse que se trataba de la misma escuela con mayor independencia y una denominación acorde a las condiciones del mercado laboral y educativo del momento.

(b) El origen en la UNI

Otro elemento importante aparece indicado en (CC, 2019), en donde indican la llegada en 1964 del primer computador a la UNI. En el documento hacen mención del papel que jugó el centro de computo de la universidad para entrenar en capacidades informáticas a sus estudiantes; algo que no era posible obtenerlas en los planes de estudio o programas académicos.

El documento (CIP, 2006) hace mención a la creación del programa de ingeniería de Sistemas de la Universidad Nacional de Ingeniería UNI en el año 1974. Se indica que este programa estuvo influenciado por el enfoque de sistemas internacional. En el mismo documento también mencionan que el objetivo no era crear especialistas en computadoras o software. Los autores reafirman esa apreciación quizás para destacar que la asimilación de un cuerpo de conocimiento de informática no llegaría al país hasta las próximas décadas. Sin embargo la necesidad de especialistas en sistemas de información ya se había instalado como oportunidad económica y académica durante los primeros esfuerzos de difusión tecnológica por parte de las casas matrices de computación. Se debe indicar que la computación con sus sub-campos bien definidos como estándares se establecería recién en la década de los 90. En el tiempo en que aparecen las profesiones en el Perú (70 y 80), esas disciplinas también se encontraban en proceso de maduración. Por ello es importante indicar que si bien es cierto que la ingeniería de sistemas se gestó y maduró con un enfoque en los

sistemas complejos sean o no informáticos. El ejercicio profesional posicionó a esa profesión en ese ámbito. Zakimi, también lo indica en (Zakimi & Reátegui, 2012, p.15)

(c) Ingeniería informática en la PUCP

La universidad Católica crea su programa de ingeniería informática en 1990 (Pucp, 2017), en un primer momento orientado a ciencia de la computación, pero al poco tiempo de su creación re-orientan a un perfil más tecnológico, algo que lo menciona uno de los protagonistas de esas decisiones Pucp (2017):

Dos años después se hizo notoria la necesidad de repensar la especialidad. La carrera estaba demasiado orientada hacia las matemáticas: “Los alumnos estudiaban algoritmos y sistemas, cuestiones complejas que si bien servían para desarrollar software de base, en ese momento no eran de mucha utilidad en el Perú”, explica el Ing. Solis. Se nombró una comisión para re-enfocar la carrera y re-evaluar el plan de estudios, la cual estuvo presidida por el Ingeniero Luis Ríos. Debían preguntarse qué tipo de profesional querían formar: “Lo que necesitábamos era un profesional que resolviera los problemas empresariales. Empresa no es solo la que hace negocios; es también el estado, la iglesia, la universidad, pues tienen que manejar sus recursos a partir de la toma de decisiones. La informática debía ayudar a eso”, comenta.

(d) Informática en la universidad Nacional de Trujillo

La universidad nacional de Trujillo crea su programa de informática en el año 1995, con un enfoque orientado a ciencia de la computación. Es destacable que la universidad resuelve la tensión del enfoque orientado a ciencia y la graduación en ingeniería, ofreciendo una graduación de bachiller en ciencias y una titulación en ingeniería. La UNITRU también tiene la carrera de ingeniería de sistemas, con componentes de estudio de informática. Lo cual no debería ser un problema, pero es interesante que una misma universidad se ofrezcan los dos perfiles predominantes de informática en el país : ingeniería de sistemas e ingeniería informática.

(e) Informática en Arequipa

La UNSA crea su primer programa de ingeniería de Sistemas en 1995, dos décadas después incorpora su escuela de ciencia de la computación en el año 2015. Ofreciendo una licenciatura en ciencia de la computación. La universidad UCSP cambia su carrera de ingeniería informática a la denominación de ciencia de la computación en el año 2006, la universidad entra en funcionamiento en el año 1997.

La tensión artificial entre ciencia e ingenierías

En las formas y diseños de creación de las escuelas se detecta una tensión entre quienes fomentan una adopción y creación de escuelas de informática con mayor preponderancia del nombre ciencia de la computación y por otro lado la adaptación de las escuelas de ingeniería a perfiles pragmáticos respecto al fin de la institución de encontrar empleabilidad para sus egresados. Sin embargo esas tensiones son producto de las rivalidades entre los colectivos y su influencia respectiva en la sociedad. La comunidad que impulsa la adopción de planes de computación con referentes busca una conexión con la red académica de informática internacional. Pero el tema no es blanco sobre negro, en el mundo también está lleno de matices y latinoamérica también tiene este tipo de tensiones y diferencias. En este estudio se plantea que la tensión puede reducirse mientras las instituciones adoptan las innovaciones de la informática como áreas, competencias o nuevos programas de estudio. El presente estudio continua las vías que el presente autor, identificó en el estudio previo (Alvarado-Revatta, 2015). En el estudio se planteó rutas de adopción de innovaciones académicas en las instituciones, se resumen de esta manera:

- 1) Reorientar o reforzar la ingeniería de sistemas existentes a estándares de ingeniería de Sistemas (INCOSE y otros).
- 2) Renombre y reforma de diseño académico y denominación de carrera. Opción elegida por UCSP (Arequipa).

- 3) Creación de nueva escuela con denominación de carrera acorde a un perfil internacional. Opción elegida por la UNI (Lima), UNT (Trujillo) y la UNSA (Arequipa).
- 4) Reforma de diseño académico sin cambio de denominación de carrera, ingeniería de sistemas o ingeniería informática con competencias de informática (Alternativa identificada en universidades extranjeras ORT de Uruguay).

En la presente investigación planteamos que la ultima opción de una reforma con innovaciones graduales de los planes de estudio sin modificar estructuras académicas es viable. Para concretar herramientas que apoyen esa posibilidad es que se desarrolla la presente investigación.

Modelos de la teoría moderna de la informática

Evolución de las sub-áreas de la informática

Peter J. Denning, es uno de los científicos informáticos más reconocidos que ha estudiado las implicancias y el alcance de la informática a través de los años. Reafirma el concepto de ciencia de la informática (Denning, 2005), así como los elementos de ingeniería de la ingeniería del software (Denning & Riehle, 2009). Denning (Denning, 2003), remarca la fuerte conexión entre la informática y la ingeniería, más cercana que la relación existente entre otras ciencias y sus correspondientes instrumentalización en ingeniería, por ejemplo la diferencia entre Química e ingeniería Química. Explica esa relación por la herencia derivada de la ingeniería eléctrica, los métodos algorítmicos diseñados para resolver problemas de ingeniería entre más ejemplo.

Para el presente estudio, se tendrá en cuenta la arquitectura del cuerpo de conocimiento que se encuentra explicado con tres paradigmas que derivan de la herencia de las matemáticas, ciencia e ingeniería; esta división propuesta por Denning nos permitirá según la propuesta del presente estudio encontrar relaciones de los planes de estudio y el cuerpo de conocimiento de la informática.

Denning, establece 3 paradigmas principales en el cuerpo de conocimiento de la informática:

- Teoría: Desarrollo de marcos conceptuales y notaciones para el entendimiento de las relaciones entre objetos en un dominio y las consecuencias lógicas de axiomas y leyes.
- Experimentación (abstracción): Exploración de modelos de sistemas y arquitecturas dentro de dominios de aplicación y validar si esos modelos pueden predecir nuevos comportamientos con precisión.
- Diseño: La construcción de sistemas informáticos que apoyen el trabajo en organizaciones o dominios de aplicación. (Denning, 2003, p.409)

La diversidad de áreas de conocimientos y sub-campos en los que se organiza el cuerpo de conocimiento de la informática puede integrar contribuciones de todos o de alguno de los paradigmas indicados. Denning plantea (es un elemento que queremos rescatar para la investigación), una tabla que organiza áreas de conocimiento versus paradigma; esta tabla que resumiremos con algunos ejemplos, es importante tomarla en cuenta pues nos permite tener una mirada más objetiva respecto a como se han organizado nuestros planes de estudio en ingeniería, de manera que esta misma tabla también nos servirá para encontrar semejanzas y diferencias en nuestras propuestas educativas.

Tabla 2.1: Sub-áreas de la informática.

Área	Teoría	Abstracción	Diseño
Algoritmos y Estructura de datos	-	-	-
Lenguajes de programación	-	-	-
Arquitectura	-	-	-
Sistemas operativos y redes	-	-	-
Ingeniería del software	-	-	-
Base de datos y recuperación de información	-	-	-
Inteligencia Artificial y robótica	-	-	-
Computación gráfica	-	-	-
Interacción persona-computador	-	-	-
Ciencia computacional	-	-	-
Informática organizacional	-	-	-

Según este esquema, ciertas áreas de conocimientos tienen componentes o innovaciones que pueden ubicarse en una de estas tres columnas; estas columnas, establecidas como paradigmas, son de especial importancia; para entender como se ha desarrollado la informática a nivel local. Nuestro sistema se ha adaptado al contexto económico

y científico del país, lo cual no significa que sea estático y que no pueda cambiar pero precisamente esos cambios no tienen la obligatoriedad de romper necesariamente los caminos o rutas elegidas por las casas de estudio durante su desarrollo. Al reconocer cada casa de estudio, que elementos o componentes de la informática desarrollan y en que sub-campo y paradigma se encuentran, podemos tener el elemento integrador entre las comunidades peruanas. A menudo las diferencias son estrictamente debates de la comunidad académica, en el campo laboral esas diferencias no son tan notorias.

Veamos algunos ejemplos de sub-campos de conocimiento y paradigmas:

Tabla 2.2: Algoritmos y estructuras de datos.

Sub-Area	Paradigma
Complejidad computacional	Teoría
Concurrencia	Teoría
Algoritmos probabilísticos	Teoría
Reconocimiento de patrones	Teoría
Algoritmos de grafos	Teoría
Divide-y-conquista	Experimentación
Programación dinámica	Experimentación
Interpretes de estado finito	Experimentación
Algoritmos aleatorios	Experimentación
Testeo de Algoritmos	Experimentación
Librerías de software	Diseño
Protocolos de comunicación	Diseño

Veamos ahora un ejemplo con la sub-área correspondiente a lenguajes de programación:

Un tercer ejemplo, adaptado de (Denning, 2003) para mostrar esta relación entre sub-área de conocimiento y paradigma, lo tomaremos para la sub-área de sistemas operativos y redes.

Estos ejemplos no son tomados al azar, por el contrario son una muestra de la evolución del cuerpo de conocimiento de la informática.

La sugerencia que podemos deducir del desarrollo actual de la informática es que tanto los programas de estudio de pre-grado de ingeniería informática, algunas ingeniería de sistemas y los programas de nueva denominación de ciencia de la computación comparten la enseñanza de estas áreas de conocimiento pero a distintos niveles de enseñanza y exploración de los contenidos; las que concentran el enfoque científico

Tabla 2.3: Lenguajes de programación

Sub-Area	Paradigma
Lenguajes formales	Teoría
Teoría de autómatas	Teoría
Maquinas de Turing	Teoría
Reconocimiento de patrones	Teoría
Teoría de tipos	Teoría
Lógica matemática	Teoría
Programación funcional	Experimentación
Lenguajes procedurales	Experimentación
OO analisis y diseño	Experimentación
Paradigmas de programación	Experimentación
Compiladores e interpretes	Diseño
Entornos de programación	Diseño
Hojas de cálculo	Diseño
Debugging-Tracing	Diseño

Tabla 2.4: Sistemas Operativos y Redes

Sub-Area	Paradigma
Teoría de concurrencia	Teoría
Algoritmos de planificación	Teoría
Maquinas de Turing	Teoría
Teoría de Colas	Teoría
Criptología	Teoría
Lógica matemática	Teoría
Gestión de almacenamiento	Experimentación
Lenguajes procedurales	Experimentación
Procedimientos remotos	Experimentación
Protocolos y servicios	Experimentación
Sistemas de tiempo compartido	Diseño
Árbol del sistema de archivos	Diseño
Controladores de dispositivos	Diseño
Capas de protocolos	Diseño

se centrarán en aspectos teóricos y de experimentación; las que se expresan como profesiones de ingeniería en los campos de experimentación y diseño.

Validar esa posibilidad va en la línea de esta investigación, sin embargo dado que tenemos poca información detallada nos concentraremos en semejanzas por nombres de cursos que podamos inferir inicialmente. Punto que revisaremos en el planteamiento de las hipótesis.

2.3 Estado del Arte

El modelo de competencias y los perfiles ACM/IEEE

Como se identifica en los antecedentes, la diversidad de denominaciones ha sido un tema de estudio e interpretación que se ha mantenido activo en el país y la región. Como lo demostró (Ramos et al., 2013), independiente de la denominación, el tipo de escuela o graduación; una propuesta educativa puede tener afinidad con los perfiles o curricula ACM/IEEE de manera que es posible establecer ciertas equivalencias. Esto basado en un estudio de competencias y sus respectivas perfiles, el estudio planteó una herramienta interesante para que las instituciones puedan visibilizar el grado de afinidad con ciertos perfiles internacionales. La propuesta plantea auditar los cursos o unidades formativas y validar el alcance o no de las competencias identificadas y agrupadas por perfiles y de ese alcance validar cuanta cercanía tiene; en el estudio se utiliza los conceptos de intensidad es decir cuantas competencias se aplican y cobertura, es decir de cada competencia cuanto.

En la propuesta se enumeran las competencias profesionales de los 5 perfiles establecidos hasta ese tiempo en las guías curriculares ACM. Se establece un listado de competencias generales y específicas. Por ejemplo:

Tabla 2.5: Competencias Generales

Competencias
Aplicar fundamentos matemáticos e informáticos
Perspectiva crítica y creativa en la identificación y solución de problemas
Identificar el papel de algoritmos y estructuras de datos
Implementar algoritmos y estructuras de datos
Aplicar principios de ingeniería del software
Comprender las limitaciones de computación

En 2.5 se muestra un extracto de las competencias generales de un profesional de informática del experimento propuesto por (Ramos et al., 2013), en el documento indican que se basan en (Shackelford et al., 2006).

En 2.6 se muestra un extracto de las competencias específicas de un profesional de informática. En el documento indican que se basan en las competencias específicas

Tabla 2.6: Competencias Específicas para informática

Competencias
Modelar y diseñar sistemas entendiendo implicaciones y alternativas
Utilizar teoría, práctica y herramientas para el proceso de un sistema
Aplicar principios de gestión eficaz, organización y habilidades de recuperación de información
Aplicar principios de HCI y construir elementos de computación gráfica
Implementar el uso eficazmente de herramientas del ciclo de desarrollo de software

para "Computer Science", la última revisión específica para informática como ciencia se encuentra en (Joint Task Force on Computing Curricula & Society, 2013).

Al revisar las competencias resumidas, puede identificarse también cierta generalidad y falta de precisión en lo que corresponde a una competencia. Es más al igual que sucede con las diferencias de competencias entre un informático y un ingeniero del software, las competencias parecerán muy similares.

Al final el proyecto es una guía de auto-evaluación, por lo que cada institución al final revisa dos variables por cada curso o componente educativo: La cobertura de la competencia y la intensidad de la competencia. La cobertura de la competencias es un indicador del porcentaje de competencias que son evaluadas parcial o totalmente dentro de una categoría. La intensidad indica el porcentaje de cursos que desarrollan total o parcialmente las competencias de una categoría.

Tabla 2.7: Resultados Intensidad y cobertura de competencias: Ingeniería en Sistemas ORT

Competencias	Intensidad	Cobertura
Generales	20%	92%
ciencia informática	17%	91.7%
sistemas de información	7%	100%
ingeniería del software	16%	100%
ingeniería de computadoras	4%	100%
tecnología de la información	6%	85.7%

En 2.7 se indica en el informe Ramos et al. (2013, p.22) que el programa de ingeniería en Sistemas de la universidad ORT de Uruguay, tiene un buen porcentaje de intensidad, en competencias generales, de ciencia informática y de ingeniería del software, en lo que corresponde a cobertura tiene un cumplimiento importante, en algunos

casos de 100%.

El proyecto Tuning Latinoamérica para informática

En (Zakimi & Reátegui, 2012) se hace un experimento para validar la importancia de las competencias entre dos grupos de interés vinculados con una propuesta educativa: Los gerentes y los docentes, para validar si ambos grupos dan el mismo nivel de importancia a un conjunto de competencias. El grupo de competencias generales y específicas tomadas por el estudio, se originan en el proyecto Tuning (Contreras Véliz et al., 2013) y las competencias específicas de (ANECA, 2004).

El proyecto Tuning da una mirada documentada a la diversidad de nuestra “geografía” de propuestas educativas de la región y también del Perú.

En el documento se resume esa diversidad con la siguiente mención: Contreras Véliz et al. (2013, p38):

En Computación e Informática los institutos ofrecen carreras técnicas y las universidades carreras de Bachiller y profesionales de Licenciado o Ingeniero. La mayoría de las titulaciones universitarias del área corresponden a “Ingeniero” y muy pocas a “Licenciado”. En Perú no existe consenso entre las universidades sobre el nombre de los títulos. En el área informática se incluyen términos como “Computación”, “Informática”, “Sistemas”, entre otros. Entre los títulos de Computación e Informática presentes en el Perú están “Ingeniería de Sistemas” (30 universidades), “Ingeniería de Sistemas e Informática” (8 universidades), “Ingeniería Informática y de Sistemas” (8 universidades), “Ingeniería Informática” (7 universidades), “Ingeniería de Computación y Sistemas” (3 universidades), “Ingeniería de Computación y Sistemas” (3 universidades) e “Ingeniería de Software” (2 universidades). Actualmente, algunas universidades ofrecen el título de “Licenciado en Ciencia de la Computación”. El año 2010 se registraron alrededor de 48.000 estudiantes en carreras de informática.

Al igual que en estudio de (Ramos et al., 2013), el proyecto Tuning, observa la diversidad no solo en el Perú sino en toda la región. Para encontrar mecanismos de

entendimiento entre las propuestas educativas, se plantean la identificación de competencias generales y específicas por especialidad.

Según la definición de metodología Tuning (project, 2006), se busca armonizar estructuras y programas educativos respetando su diversidad y autonomía. Este enfoque difiere en parte del proceso de establecer un filtro por competencias específicas de perfiles según un modelo pre-determinado por guías o curriculas. Por lo que el proyecto Tuning busca crear un meta-perfil profesional del área informática en la región. El proyecto busca identificar cualidades fundamentales independientes del área de especialización o del contexto de formación.

Veamos una muestra de las competencias, tomado de Contreras Véliz et al. (2013, p.40)

Tabla 2.8: Competencias genéricas del informe Tuning

Competencias	Importancia
Capacidad de aplicar los conocimientos en la práctica	3.77
Capacidad de abstracción, análisis y síntesis	3.76
Capacidad para identificar, plantear y resolver problemas	3.76
Capacidad para para aprender y actualizarse permanentemente	3.75
Capacidad de trabajo en equipo	3.66
Capacidad de investigación	3.59
Capacidad para formular y gestionar proyectos	3.59
Responsabilidad social y compromiso ciudadano*	3.09

En 2.8 no se indican todas las competencias genéricas, pero es una muestra que permite observar el grado de síntesis necesaria para encontrar mecanismos de armonización de las propuestas educativas de la región.

Las competencias del meta-perfil se resumen en la siguiente imagen, tomada de Contreras Véliz et al. (2013, p.45):

El proyecto Tuning aborda el enfoque de competencias como elementos que sirvan de comunicación entre las propuestas educativas y los grupos de interés que influyen o participan del proceso formativo. El aporte es relevante para encontrar patrones que puedan generalizar las propuestas educativas que al detalle se muestran divergentes.

Tabla 2.9: Competencias de la disciplina

Competencias	Importancia
Aplicar el conocimiento de ciencias de la computación, de tecnologías de la información y de las organizaciones, para desarrollar soluciones informáticas	3.77
Concebir, diseñar, desarrollar y operar soluciones informáticas basándose en principios de ingeniería y estándares de calidad	3.72
Aplicar estándares de calidad en el desarrollo y evaluación de soluciones informáticas	3.58
Identificar oportunidades para mejorar el desempeño de las organizaciones a través del uso eficiente y eficaz de soluciones informáticas	3.55
Asimilar los cambios tecnológicos y sociales emergentes	3.53
Aplicar el enfoque sistémico en el análisis y resolución de problemas	3.52
Comprender y aplicar los conceptos éticos, legales, económicos y financieros para la toma de decisiones y para la gestión de proyectos informáticos	3.45
Desempeñar diferentes roles en proyectos informáticos, en contextos multidisciplinarios y multiculturales, tanto locales como globalizados	3.42

2.4 Búsqueda de semejanzas entre propuestas internacionales y la adopción regional

La adaptación a latinoamérica se encuentra en los proyectos de (Ramos et al., 2013) donde se busca cierta adaptación de las competencias ACM/IEEE a la región, aunque

Componentes del Perfil de Egreso del Área Informática
Tuning América Latina



en ese estudio se presenta una traducción de las competencias de las guías ACM 2005 (Shackelford et al., 2006), no se toma en cuenta en realidad el contexto sino una valoración por parte de la institución del cumplimiento o no de las competencias. Otro enfoque más consensuado se encuentra en (Contreras Véliz et al., 2013) donde se busca identificar por expertos regionales las competencias genéricas y específicas de profesionales de informática, es decir primero identificar las competencias de cada contexto y luego identificar cierta afinidad y generalidad.

En la presente investigación, usaremos como modelo de guías internacionales a los referentes ACM68 (Atchison et al., 1968), ACM2001 (The Joint Task Force on Computing Curricula, 2001), ACM2013 (Joint Task Force on Computing Curricula & Society, 2013), la idea es habilitar el modelamiento de los cambios en los referentes internacionales en particular de las áreas de conocimiento que incluyen estos documentos.

Las relaciones entre contenidos y áreas basada en grafos

En la línea de encontrar similitudes y relaciones entre planes de estudio de informática, una línea de investigación utiliza conceptos de grafos para identificar relaciones tanto de áreas de conocimientos como de competencias y usar las propiedades de la teoría de grafos para mostrar esta información. En la investigación de (Pasterk & Bollin, 2017) el objetivo es encontrar similitudes entre curriculas de informática para educación escolar. Los autores encuentran valorable el modelamiento con grafos de las curriculas. Para la presente investigación, utilizaremos ese enfoque para modelar las adopciones y cambios en las curriculas de la oferta educativa peruana en

informática pero centrados en la oferta universitaria.

Modelo basado en grafos

En (Sedgewick & Wayne, 2011) se define un grafo como:

un conjunto de vértices y una colección de enlaces donde cada enlace conecta un par de vértices.

Se puede representar las propuestas educativas o planes de estudio así como los referentes o guías estándar en un modelamiento con grafos.

En (Pasterk & Bollin, 2017) los autores plantean un modelamiento que permita tener en cuenta elementos de conocimiento y competencias vinculados con los estándares. Entre el proceso de modelamiento utilizan texto para hacer anotaciones en las relaciones de los grafos, por ejemplo mencionan identificar vértices y enlaces, donde los enlaces representan relaciones “de requerido por” y “expandido por” para áreas de conocimiento. Los autores introducen el uso de conceptos de la teoría de grafos como métricas para poder realizar comparaciones.

La investigación de (Marshall, 2012) modela las curriculas ACM/IEEE como grafos y realiza comparaciones entre los estándares ACM/IEEE 2001, 2008 y 2013, además también hace una comparación con un plan de estudios del mundo real. Los resultados se centran en una métrica basada en comparar el número de nodos y enlaces respecto a cada modelo.

El aporte de la presente investigación parte de esa línea de investigación e introduce el uso de un algoritmo de máximo común sub-grafo y de las medidas de distancia basado en isomorfismo para realizar las comparaciones respecto a programas de estudio y los referentes. Es decir introducir una métrica de similaridad de grafos, además de aplicar la comparación entre modelos ACM 68 (Atchison et al., 1968), 91 (acm, 1991), etc.

Por lo que nuestro alcance es mayor en el sentido de intentar responder preguntas de relación en el tiempo con los referentes internacionales; un factor importante para estudiar el desarrollo curricular de nuestras instituciones.

En (Lightfoot, 2010) se propone el modelamiento de curriculas como grafos de modo que puede estudiarse métricas como elementos para el análisis. Para el presente análisis se considera relevante identificar un sub-grafo que represente los elementos centrales del área de informática y buscarlo en los grafos de planes de estudio de las universidades que usaremos en nuestro estudio.

2.5 Contribución teórica para el proyecto

Similaridad de grafos

En (Koutra et al., 2011), se identifica el marco conceptual de algoritmos de similaridad. El teorema para el algoritmo es el siguiente:

Dado dos grafos $G_1(n_1, e_1)$ y $G_2(n_2, e_2)$, con posiblemente diferente número de vértices y aristas, y la red que se forma entre los vértices.

Encontrar, un a) algoritmo que calcule la similaridad de los dos grafos y que retorne b) una medida de similaridad (un valor real entre 0 y 1).

Concepto de isomorfismo y sub-grafos

En (Ullmann, 1976) se define un isomorfismo de la siguiente manera:

Un sub-grafo G es un grafo cuyos nodos y enlaces se encuentran en G . un grafo G_α es isomorfo a un sub-grafo de un grafo G_β si y solo si existe una correspondencia 1:1 entre el conjunto de nodos del sub-grafo y G_α preserva sus relaciones.

En (Bunke, 2000) indica que el isomorfismo de sub-grafos puede ser usado para encontrar si un objeto es parte de otro, o si un objeto está presente en un grupo de objetos. Se menciona el concepto de máximo común sub-grafo, que puede ser usado para medir la similaridad de objetos incluso si no existe isomorfismo de grafos o sub-grafos entre los grafos correspondientes. Se busca encontrar el sub-grafo máximo en común entre dos grafos, el mayor representa la similitud.

Una representación teórica de isomorfismo de grafos la encontramos en (Gao et al., 2010):

Sea $G_1 = (V_1, E_1, \alpha_1, \beta_1)$ y $G_2 = (V_2, E_2, \alpha_2, \beta_2)$ dos grafos. Un isomorfismo de grafos entre G_1 y G_2 es un mapeo biyectivo $f : V_1 \rightarrow V_2$ tal que:

1. $\alpha_1(x) = \alpha_2(f(x)), \forall x \in V_1$
2. $\beta_1((x, y)) = \beta_2(f(x), f(y)) \forall (x, y) \in E_1$

Si $V_1 = V_2 = \emptyset$, entonces f es llamada un grafo vacío isomorfo.

Máximo común sub-grafo

De (Bunke, 2000) y (Gao et al., 2010) se define lo siguiente: Sea G_1 y G_2 dos grafos y $G'_1 \subseteq G_1$, $G'_2 \subseteq G_2$. Si existe un grafo isomorfo entre G'_1 y G'_2 , entonces ambos G'_1 y G'_2 son llamados sub-grafo común de G_1 y G_2 . Si no existe otro subgrafo en común de G_1 y G_2 que tenga mas nodos que G'_1 y G'_2 . G'_1 y G'_2 son llamados un MCS de G_1 y G_2 . Dentro del concepto de común sub-grafo existen variantes como a) Máximo común sub-grafo inducido el cual lo hemos explicado, la búsqueda de la mayor cantidad de vértices, b) En máximo común sub-grafo de enlaces, que permite encontrar el grafo la mayor cantidad de enlaces y que sea isomorfo a ambos grafos comparados.

Como lo indica (Bunke, 2000) en usos del mundo real a veces no existe una coincidencia exacta por lo que se considera un grado de error de tolerancia, usando lo que conocemos como: distancia de edición de grafo. Una operación de edición puede ser una eliminación, inserción o sustitución. Estas operaciones pueden ser aplicadas a nodos o enlaces. La distancia de edición de dos grafos g y g' es definida como la secuencia mas corta de operaciones de edición que transforman g en g' , la secuencia mas corta indica la mayor similaridad entre los grafos. También indican que algunas operaciones de edición pueden tener mas importancia que otras. Por lo que las operaciones de edición tiene un costo asociado individualmente, en nuestro caso no consideraremos un costo asociado para calcular la distancia.

Tanto los conceptos de isomorfismo, de sub-grafos isomorfo y del máximo común sub-grafo son todas instancias de un computo de la edición de distancia de grafos, considerando funciones de costo por operación.

Métrica de similaridad entre grafos

Según (Horst Bunke, 1998) y (Bunke, 2000) define una métrica que no considera las costosas operaciones de edición para convertir un grafo g en g' sino que establece una medida usando el concepto de Máximo común sub-grafo. La medida se establece como la distancia entre dos grafos, usando el número de vértices de un grafo como $|g|$. La fórmula permite cuantificar la distancia y complementando a 1 se obtiene la similaridad entre dos grafos.

$$\delta(g, g') = 1 - \frac{|mcs(g, g')|}{\max(|g|, |g'|)}$$

Este indicador basado en el cálculo del número de nodos del máximo común sub-grafo y el máximo entre grafos comparados nos dará un valor entre 0 y 1 que indica el grado de similitud entre ambos grafos comparados. A menor valor indica una mayor similitud. Es decir si el valor se acerca a 1 son grafos disimiles, por el contrario si se acerca a 0 son grafos similares.

Para efectos de cuantificar la similaridad definiremos a Z como el complemento a 1 del valor de δ :

$$\zeta = 1 - \delta(g, g')$$

Este valor lo usaremos para medir entre 0 y 1 donde un valor cercano a 0 implica poca similaridad y un valor cercano a 1 mayor similaridad. En el estudio usaremos un valor $\zeta > 0.2$.

Algoritmo de Máximo común sub-grafo

El algoritmo de máximo común sub-grafo es un problema NP por lo que su ejecución se complica conforme crece el número de vértices de los grafos a comparar. El problema es un área de estudio propiamente dicho y de mucha aplicación en el mundo de la química informática. Tiene distintos algoritmos e implementaciones desde la década de los 70 hasta la actualidad. En la presente propuesta usaremos una implementación

que identifica el máximo común sub-grafo conectado y usaremos ese valor para el calculo de similitud entre los programas de estudio.

La decisión para elegir el algoritmo que mostraremos se fundamenta en principios pragmáticos que nos permitan alcanzar los resultados experimentales de la solución al problema; no es el propósito de la investigación evaluar el método de implementación del algoritmo de máximo común sub-grafo, considerando que el número de nodos a comparar en nuestros ejemplos es relativamente pequeño por lo que un tiempo de ejecución optimo no es significativo.

A continuación se explica de modo general una modelo de algoritmo que usamos en el presente proyecto. En esta implementación se considera a los sub-grafos conectados pues permite encontrar la estructura de relaciones comunes de mayor alcance.

Algoritmo 1: Maximo componente conectado

Input: $MCS(G, H)$

Data: G y H dos nodos.

Result: $(MCS - C)$ Maximo componente conectado

Data: Grafos G y H

$S = \text{nuevoGrafo}()$ // Crea nuevo grafo S

for $E(n1, n2)$ *en* G **do**

if H tiene enlaces $E(n1, n2)$ **then**

$S.add(E(n1, n2))$ // Agrega enlace E a S

$\text{Componentes} = \text{ComponentesConectados}(S)$

$\text{ComponenteMasGrande} = \max(\text{Componentes})$

Result: $\text{ComponenteMasGrande}$

Esta implementación corresponda a un aporte pragmático de (Magin, 2019) para encontrar sub-grafos máximos conectados. En la adaptación se ha usado la librería NetWorkX (NetWorkX, 2022) que internamente utiliza funciones de búsqueda de subgrados.

Los algoritmos de búsqueda de sub-grafos son ampliamente usados en el campo de la informática química o quim-informática, para encontrar patrones químicos, o patrones de moléculas. En nuestro caso, lo aplicamos para encontrar sub-grafos comunes entre guías o planes de estudio, objeto de estudio de la presente investigación.

Cuantificación de relaciones y nodos

Un dato importante que también puede considerarse en el objetivo de cuantificar la semejanza o distancia entre las comparaciones, es el uso de una operación simple de conjuntos entre los elementos a comparar. Dado un grafo g y g' calcular cuantos nodos y enlaces son comunes sin que un nodo existente en g esté presente en g' o viceversa. Este simple número puede ayudar a visibilizar los elementos en común entre ambos grafos.

Algoritmo 2: Grafo con elementos comunes a G y H

Input: $Comun(G, H)$

Data: G y H dos nodos.

$T = \text{nuevoGrafo}()$

```

for  $E(x,y)$  en  $H$  do
    if  $G$  tiene  $E(x,y)$  then
         $T \text{ add}(E(x,y))$ 

```

Result: T

Estos valores son importantes para medir la proporción de cursos comunes en ambos grafos de la comparación. También puede servir para medir la proporción de áreas de conocimiento que corresponda en cada curso comparado.

HIPOTESIS Y VARIABLES

3.1 Hipotesis General

Como se ha identificado existe por lo menos dos enfoques para identificar relaciones de los planes de estudio con el objetivo de incorporar innovaciones en las mismas sobre la influencia de curriculas internacionales. Estos dos enfoques son: a) Áreas de conocimientos y b) Competencias.

La presente investigación pretende encontrar información esencialmente en el modo de áreas de conocimientos de los planes de estudio de sistemas e informática de universidades de una pequeña muestra, la hipótesis central propone identificar si existe similitud entre los planes de estudio de distintas escuelas independiente del nombre de la carrera y si se asemejan a planes o guías internacionales independiente de la fecha o etapa de evolución.

La identificación de estas relaciones de áreas de conocimiento requerirá cierto tratamiento de la información de los planes de estudio y de las curriculas internacionales. Se limitará el alcance de la información usada y se aplicará un método para agrupar los cursos similares o pertenecientes a un mismo cuerpo de conocimiento con un solo identificador o nombre que nos permitirá reconocer como el mismo elemento independiente del plan de estudio o de la guía a comparar.

En este punto existe cierto grado de flexibilidad que se establece en la codificación del curso o área de conocimiento. Como primer alcance se utilizará información pública de los planes de estudio así como información pública de guías internacionales. La información sobre cursos y sus dependencias se modelarán como grafos y enlaces.

Se considera el uso del algoritmo Máximo Común Subgrafo así como operaciones de proporcionalidad, para cuantificar y visualizar relaciones o posibles relaciones entre programas de estudio.

Nuestra hipótesis general la podemos expresar de la siguiente manera:

Dado un modelo en grafos de un plan de estudios o una guía internacional se identifica una relación entre un plan de estudios G_1 y G_2 como la métrica entre ambos modelos usando el concepto de similaridad grafos. De modo que la similaridad es significativa.

Definiremos un modelamiento siguiendo los siguientes pasos:

1. Se considera el marco teórico de sub-grafos y coincidencia de gráficos para modelar los planes de estudio y/o curriculas (Referentes), reproduciendo una comparación usando como métrica la similaridad entre subgrafos. Visualizaremos también el máximo común subgrafo conectado como un elemento de similitud entre planes y/o curriculas, así como las relaciones en común para identificar las similitudes o diferencias.
2. La data será depurada y/o preparada para centrarse en los cursos en contexto informático y de matemáticas, se descarta cursos de gestión, ciencia general y sociales a pesar que son temas que influyen en las competencias modernas, sin embargo lo que se busca es encontrar la similaridad en áreas centrales de la disciplina informática en las implementaciones académicas peruanas.
3. Se modela los planes o guías que usaremos para el alcance de la investigación en grafos y se usará un diccionario de nombres que agrupen áreas comunes o cursos similares a esto lo denominamos establecer un índice. Se crea una Tabla de Comparación. Tanto El Plan como el Referente tendrán una tabla de comparación que permita identificarlos usando un nemónico (índice) que represente el curso o área de conocimiento que abarca la oferta educativa.
4. Se aplicara los algoritmos a la codificación de los planes y guías, se realizará las comparaciones usando dos modelos: modelo 1) se identifica los cursos de los planes de estudio con equivalencias a ejemplos de curriculas internacionales, en caso de curriculas entre universidades se crea un Id y nombre común a los cursos equivalentes y se modelan los grafos usando ese identificador. En el modelo el mapeo de relaciones se respeta la matriz de relaciones original de la curricula internacional. En el modelo 2) Se ambos grafos mantienen sus

relaciones originales. Se presume que bajo esa premisa el subgrafo conectado sea menor o incluso no se pueda identificar.

5. Cuantificar el número de nodos y relaciones por cada sub-grafo de comparación y tabular los resultados. Estos datos nos permitirán cuantificar tanto los cambios en las curriculas como las similitudes entre los planes existentes y las propuestas educativas locales. Cuantificar los enlaces comunes entre ambos grafos para visibilizar la relaciones.
6. Mostrar la evidencia numérica que apruebe o rechace la hipótesis general y secundarias.

El alcance de esta investigación solo considera las áreas de conocimiento en común entre las propuestas educativas de ingeniería y la informática y planes internacionales o comparación entre planes de una o entre instituciones. No se aborda el campo de las competencias pues no hay información pública disponible. En el caso de las áreas de conocimiento, se compararan a nivel de cursos, estos se infieren de revisión de planes de estudio de los cursos que tengan información pública.

3.2 Hipotesis Específicas

A continuación usaremos la hipótesis principal para responder preguntas vinculadas con el tema que afectan el proceso de desarrollo de las profesiones universitarias en informática. Esta pregunta principal la podemos desagregar en las preguntas secundarias que generan las hipótesis específicas.

1. P2. (H2) Dado un grafo G_1 que representa una profesión de ingeniería de sistemas u otra propuesta existente en la oferta peruana y G_2 un referente internacional de ACM/IEEE u otro. Se identifica que la similaridad es significativa.
2. P3. (H3) Dado un grafo G_1 que representa el plan de estudio de una carrera vinculada a la informática de un año X y G_2 un plan de la misma carrera pero del año Y. Se Identifica que la similaridad es significativa. Una modificación también podría permitir comparar dos carreras locales de distintos años y validar la similaridad.

Dado que usaremos una métrica basada en el Máximo común subgrafo, el concepto de significancia pasa por establecer un valor que indique la presencia o no de similaridad. El concepto de similaridad se definió en el capítulo 2 2.5 , en este caso tomaremos un valor $\zeta > 0.2$ como el valor que indique o no la presencia de significancia de similaridad. Es decir una métrica que supere ese valor indicaría cierto grado de similaridad, uno que sea menor, indicaría una similaridad no significativa.

Otro elemento que estamos utilizando es 2.5 la proporción en una de los planes o guías respecto a la cantidad de nodos en común. En este caso usaremos una proporción 0.1 para indicar cierto grado de similaridad por nodos.

3.3 Identificación de variables

Se define las siguientes variables para el presente estudio.

Plan: Un plan de estudios estará compuesto por unidades de enseñanza que serán cursos que representan parte o un área de conocimiento. Se establecerá que cada plan de estudio representa un grafo donde cada nodo será etiquetado con un identificado que a su vez representa un nombre de un área o sub-área de conocimiento dentro de la informática.

Un Referente: Un referente es similar a un plan de estudios pero obtenido de los modelos propuestos en los referentes internaciones. Estos son un conjunto de acuerdos de áreas de conocimientos que representan los consensos para orientar la enseñanza en la informática. Cada referente tiene un tiempo de creación y una o varias instituciones tutelares. Se modela usando como datos un nombre que será común y con este nombre se buscará la equivalencia en caso se realice comparaciones.

Un curso: Un curso será una propuesta educativa que aparece en un Plan o Referente y que aborda un conjunto de *UnidadConocimiento* que corresponde a un mapeo estandarizado de unidad de conocimiento que pertenece a la clasificación de una guía internacional. Un curso estará formado por varios nodos cada uno de ellos es una *UnidadConocimiento*

Máximo común subgrafo (MCS): Es un sub-grafo que representa relaciones y cursos que conforman un plan o un referente, representa lo común a ambos. El subgrafo re-

sultante tendrá dos valores que nos pueden dar información respecto a los elementos comunes entre comparaciones. Este se representará como una tupla de dos valores (*numNodos*, *numRelaciones*). En sentido práctico usaremos para identificar un subgrafo el concepto de subgrafo conectado para poder encontrar el elemento más grande común a dos comparaciones.

Distancia de grafos: Es una medida entre 0 y 1 que a menor medida indica la similitud entre ambos grafos. Considerando subgrafos inducidos conectados o no. Tanta la distancia como el MCS están vinculados por lo que la librería o función que usemos para calcular la distancia deberá elegirse adecuadamente para comparar y encontrar grafos inducidos conectados o no. En la experimentación se explicará explícitamente esa diferencia. La distancia y similitud son dos valores complementarios, a menor distancia significa que los grafos son más similares por lo que la medida de similitud se considera complementaria a 1 del valor de la medida de la distancia entre los grafos.

3.4 Operacionalización de variables

Tabla 3.1: Matriz de operacionalización de variables

Tipo de Variable	Variable	Definición conceptual	Valores finales	Escala	Naturaleza
Independiente	Plan	Programa o plan de estudios	Grafo	Tupla	Cuantitativa
Independiente	Referente	Ejemplo de plan desde Referente	Grafo	Tupla	Cuantitativa
Independiente	Curso	Ejemplo de curso con Unidades de Conocimiento	Grafo	Tupla	Cuantitativa
Dependiente	MCSC	Subgrafo común	Grafo	Tupla	Cuantitativa
Dependiente	Similitud	Una métrica resultante	Valor real	Numero	Cuantitativa
Dependiente	Nodos Comunes	Una proporción resultante	Valor real	Numero	Cuantitativa

En 3.1 se indica la tabla de operacionalización de las variables del estudio. Los planes y los referentes se van a modelar como grafos etiquetados, siendo el identificador un nombre que será el resumen de un conjunto de nombres comunes para curso o

áreas de conocimientos. El objetivo es bajo juicio experto y usando trabajos previos poder organizar la identificación de los grafos. Por ejemplo si un curso en un plan de estudios de la universidad A se llama Sistemas Operativos, y otro curso en una universidad B se llama Sistemas Operativos y Redes, en el modelamiento se llama Sistemas Operativos que equivale a ambos cursos. Es posible poder agrupar con un solo nombre los 2 cursos como Sistemas Operativos, aún cuando podría existir diferencias y enfoques desde una concepción teórica o aplicada. Ese detalle se deberá revisar para evitar errores en el proceso de modelamiento.

3.5 Matriz de consistencia

Tabla 3.2: Matriz de consistencia

Problema	Objetivos	Hipotesis	Variables	Método
Problema Principal ¿Que relaciones podemos identificar en los planes de estudio de programas de informática con otros planes o guías internacionales?	Objetivo General Adaptar y aplicar conceptos de redes y similaridad de grafos en el análisis de relaciones entre programas de estudios y guías de perfiles profesionales.	Hipótesis General Identificación de relación entre grafos de modo que supere un umbral de similaridad.	Variables Planes, Guia, Cursos	Tipo de investigación Aplicación de algoritmos de grafos y similaridad.
Problema Secundario 1. ¿Que relaciones existe entre los programas de estudio y referentes internacionales en distintos periodos de tiempo? 2. ¿Que relaciones existe entre los planes de estudio en distintos periodos de tiempo o distintas casas de estudio?	Objetivos Secundario 1. Encontrar similaridad entre planes y guías de perfiles. 2. Encontrar similaridad entre planes en distintos periodos o casas de estudio.	Hipótesis Secundaria Identificación de relación entre grafos de modo que supere un umbral de similaridad. 1. G_1 ing. de sistemas o informática y G_2 guía internacional de ACM/IEEE. 2. G_1 que representa el plan de estudio de una carrera vinculada X y G_2 un plan de la misma carrera pero del año Y.	Variables Secundarias 1. G_1, G_2 , Identificar similaridad 2. G_1, G_2 , Identificar similaridad	Método 1. Aplicación de algoritmo MCS 2. Aplicación de algoritmo MCS

METODOLOGÍA

4.1 Tipo de Diseño de Investigación

La presente investigación se concibe como una investigación experimental en donde se hará uso de una técnica de modelamiento basada en las propiedades de los grafos para encontrar evidencia de relaciones entre los elementos conceptuales en este caso los datos considerados comparables en planes de estudio, guías internacionales o cursos. Los grafos que representan de modo independiente programas académicos y referentes internacionales, servirán para aplicar mecanismos de operación con grafos y lograr valores cuantificables de manera que sigamos a pesar de las diferencias nominales se pueda identificar relaciones cuantitativas entre los programas de estudio y referentes internacionales.

La investigación se inicio de modo exploratorio considerando a modo conceptual que han existido estudios de modelamiento de los planes de estudio y su representación en grafos, a su vez los proyectos que han abordado la problemática de la región se han centrado más en buscar la cercanía de los programas de estudio a referentes internacionales o a encontrar semejanzas en las competencias versus los referentes internacionales. Pero pocos estudios se han dado en tratar de buscar similitudes entre los programas locales sin importar el nombre, es decir la búsqueda de similitud tal cual se hace en la industria cuando contratan egresados de la diversa oferta educativa. Aunque se asume que existe tal semejanza en la oferta, no existe estudios que encuentren esa similitud como la que se propone en el presente proyecto. La investigación también es experimental y orientada al modelamiento en el sentido que la información sobre planes de estudio se modela como grafos no dirigidos para representar unidades que corresponden a curso o unidades de conocimiento que nos permitirá contar con estructuras que luego podremos comparar.

4.2 Unidad de análisis

La unidad de análisis de este estudio son planes de estudio o de referentes internacionales de la enseñanza de carreras de pre-grado en computación, informática o sistemas, sean licenciaturas o ingenierías. En especial de programas de la oferta académica peruana y en especial de la variada y diversa forma de desarrollo de las propuestas educativas peruanas. La estructura y diseño del desarrollo de la academia peruana se ha desarrollado en distintas orientaciones, en la investigación exploratoria encontramos conceptos asociados con el desarrollo de la innovación educativa y el surgimiento en el país de las tendencias educativas. En la parte experimental de esta investigación, usaremos la información de los planes para construir una representación en grafo no dirigido de unidades de conocimientos que tenga correspondencia con estos planes de estudio. El plan de estudio o el referente o guía internacional corresponde a la arquitectura de conocimientos que se imparten en la formación. Como hemos identificado la formación moderna de cualquier disciplina educativa tiene dos componentes importantes el conocimiento impartido y las competencias que se desarrollan. En este estudio usaremos a los planes y guías internacionales como elementos de estudio para identificar la arquitectura del conocimiento impartido, el estudio de competencias no se considera como parte de la investigación.

4.3 Población de estudio

La población de nuestro estudio es cualquier plan de estudios de profesiones vinculadas a la informática que se imparte en la educación universitaria del país. Los usuarios de esa información a menudo la utilizan para tomar decisiones sobre que carrera elegir o que cursos son importantes dentro de la propuesta. Los planes de estudio también cambian en el tiempo y son dinámicos por que reflejan por un lado la interpretación institucional de la propuesta pero por otro lado también son un reflejo del enfoque y de las posibilidades de una institución. Por lo tanto son una fotografía del momento de la institución y pueden servir como unidades de estudio.

Las personas que puede tener interés en este estudio va desde académicos que evalúan el progreso o desarrollo de los planes de estudio; estudiantes postulantes a las

carreras y que se enfrentan a dilemas de decisión sobre que carrera elegir; a estudiantes de las carreras que llevan matriculados para conocer las similitudes o diferencias que existen respecto a su formación y las impartidas por otros centros de enseñanza. Finalmente puede ser de interés para aquellos que estudian el desarrollo y evolución de la educación en informática del país.

4.4 Tamaño de la muestra

En el caso de la presente investigación, usaremos información de las siguientes universidades ubicadas en ciudades de marcada tendencia de cambios en la oferta educativa:

Tabla 4.1: Matriz de planes de estudio considerados en el estudio

Carrera	Universidad	Año plan
Ingeniería informática	Universidad Católica	2020
Ingeniería de Sistemas	Universidad San Marcos	2009
Ingeniería de Sistemas	Universidad San Marcos	2017
Ingeniería de Sistemas	Universidad de Ingeniería	2018

Tabla 4.2: Matriz de guías internacionales

Perfil	Organización	Graduación	Año guía
Informática	ACM	Pre-grado	1968
Informática	ACM/IEEE	Pre-grado	2003
Informática	ACM/IEEE	Pre-grado	2013

4.5 Selección de la muestra

Las muestras seleccionadas corresponden a un grupo de 4 propuestas de distintas universidades y con distintos nombres en la propuesta educativa y de 4 modelos de planes de estudio ejemplo derivados de los referentes internacionales de distinta época. La idea central es poder modelar estas propuestas para alcanzar una representación de grafos que nos permita obtener información cuantitativa del modelo.

Se ha buscado en la oferta pública de las universidades los documentos que resumen los planes de estudio vigentes a la fecha de la realización de la investigación. Se han descargado los documentos públicos y se almacenan en carpetas independientes

por universidad. La selección podría ser aleatoria pero dada la poca disponibilidad de recursos de base de datos estandarizados para los planes se ha optado elegir propuestas de los perfiles comunes en la oferta educativa: Ingeniería informática, ingeniería de sistemas, ciencia de la computación.

De igual manera con respecto a los perfiles internacionales que sirven de referentes para innovar o estandarizar los programas de estudio se van a considerar a los marcos referenciales más reconocidos a nivel internacional.

4.6 Técnicas de recolección de datos

La recolección de datos ha requerido la búsqueda de documentos oficiales publicados en Internet por las instituciones rectoras tanto de las universidades que ofertan programas de estudios como de los referentes internacionales. Las listas se encuentran indicadas en 4.1 y 4.2. La investigación se realizará analizando los documentos y creará una base de datos de términos con una tabla llave-valor donde podremos uniformizar unidades de conocimientos o cursos para luego tener como unidad de comparación y modelamiento en grafos.

A continuación se muestra el listado de actividades requeridas para la recolección de datos:

1. Revisión documental sobre innovación en sistemas sociales
2. Revisión documental sobre hechos históricos del desarrollo de la educación en informática.
3. Identificar planes de estudio de universidades que se tomaran como parte de los experimentos.
4. Identificar guías internacionales que se usaran como parte del experimento.
5. Identificar las propiedades de la teoría de grafos que sirva para la investigación.

Como se ha indicado previamente la información de estos programas se modelarán en grafos para estudiar propiedades de estos. De modo que permitan responder las preguntas de la investigación.

RESULTADOS Y DISCUSIÓN

5.1 Análisis e interpretación de la información

Proceso de clasificación, registro y codificación de los datos

En el presente estudio se utiliza datos que corresponden a planes de estudio de algunas universidades del país y de modelos de planes de estudio que representan una implementación de guías internacionales.

Esa decisión se da ante la premisa que el contenido de un curso aborda las áreas de conocimiento esperado de un programa de estudios en informática. Por ejemplo, un curso de sistemas operativos (SO) se presume comparable entre distintas propuestas universitarias y ejemplos de guía internacional.

Esa es una primera aproximación que no debe ser descartada aún si los enfoques o niveles de alcance de cada curso sean distintos, algo que se podrá analizar a futuro, cuando se compare el contenido de áreas de conocimiento de cada curso y los respectivos ejemplos de cursos de las guías.

Con esa premisa o función comparable podremos realizar una codificación con un mismo (*ID, NOMBRE*) que nos permita hacerlos identificables entre los planes de estudio y guías. De esa manera podremos encontrar relaciones a nivel de grafos.

A continuación utilizamos solo unas muestras de planes y guías para mostrar el proceso de codificación. El detalle se encuentra en el enlace del anexo A.1 que aparecerá en el anexo del documento.

Tabla 5.1: Guia ACM 68

Id	Nombre	Relacion	NombreCurso
1	B1	-	Introduction to Computing
2	M1	-	Introduction to Calculus
3	B2	(3,1)	Computer and Programming
4	M3	(4,2)	Linear Algebra
5	M2	(5,2)	Mathematical Analysis I
6	B3	(6,1),(6,5),(6,4)	Introduction to Discrete Structures
7	B4	(7,4)	Numerical calculus
8	M2p	(8,2)	Probability
9	M4	(9,5),(9,4)	Mathematical Analysis II
10	I1	(10,5),(10,6)	Data Structures
11	I2	(11,3),(11,6)	Programming Languages
12	I3	(12,3)	Computer Organization
13	M7	(13,8)	Probabilty and Statistic
29	M6	-	Algebraic Structures
14	M5	(14,8)	Advance Multivariable Calculus
15	I4	(15,10),(15,11)	Systems Programming
30	I5	(30,10)(30,11)	Compiler Construction
16	I6	(16,12) (16,6)	Switching Theory
17	I7	(17,6)	Sequential Machine
25	I8	(25,9)	Numerical Analysis I
26	I9	(26,9)	Numerical Analysis II
18	A1	(18,10)(18,11)	Formal Languages and Syntactic Analysis
27	A2	(27, 12)	Advance Computer Organization
28	A3	(28, 6), (28, 11)	Analog and Hybrid Computing
19	A4	(19,14)(19,13)	System Simulation
20	A5	(20,9)	Information Organization and Retrieval
21	A6	(21,9)	Computer Graphics
22	A7	(22,6) (22,17)	Theory of computability
23	A8	(23,19)	Large Scale Information Processing System
24	A9	(24,9)	Artificial Inteligence and Heuristic programming

Tabla 5.2: Guia ACM 2001

Id	Nombre	Relacion	NombreCurso
1	cs101	-	Programming Fundamentals
2	cal1	-	Calculus I
3	cs102I	(3,1)	The Object-Oriented Paradigm
4	cs115	(4,2)	Discrete Structures for Computer Science
5	cal2	(5,2)	Calculus II
6	sc120	-	Science course I
7	cs103i	(7,3)	Data Structures and Algorithms
8	cs210	-	Introduction to Computer Organization
9	sc	-	Science Course II
10	ps	(10,5)	Probability and Statistics
11	cs210	(11,4)	Algorithm Design and Analysis
12	cs220	(12,3)	Computer Architecture
13	am	(13,5)	Advanced Mathematics elective
14	cs225	(14,12)	Operating Systems
15	cs280	-	Social and Professional Issues
16	cs230	(16,14)	Net-Centric Computing
17	cs262	-	Information and Knowledge Management
18	cs290	(18,4)	Software Development
19	cs115	-	Discrete Structures for Computer Science
20	cn1(e)	(20,2)	Numerical analysis
21	cs302(a)	(21,10)	Probability and Statistics
22	cs270T	(22,4)	Databases
23	CS307(a)	-	Simulation and Modeling
24	CS340	-	Compiler Construction
25	GV11(e)	-	Computer vision
26	CS306	-	Operations Research
27	cs222	(27,12)	Architectures for Networking and Communication
28	cs250	(28,3)	Human-Computer Interaction
29	cs260	(29,4)	Artificial Intelligence
30	IM10(e)	-	Data mining

Tabla 5.3: Guia ACM 2013

Id	KA	Nombre	Relacion	NombreCurso
1	-	MAT171	-	Calculus I
2	-	CSC221	-	Introduction to Programming
3	AL	CSC207	(3,2)	Object-Oriented Programming
4	AR	CS552	(4,2)	Intro to Computer Architecture
5	AR	COSCMat201	(5,1)	Modeling and Simulation
6	CN	CS175	-	Computer Graphics
7	DS	MAT267	(5,1)	Discrete Mathematics
8	AL	CSC140	(8,7)	Algorithms
9	IAS	CS475	(9,3)	Computer Systems Security
10	IM	CSU-DS	(10,4)	Database Systems
11	IS	UH-AI	(11,7)	Artificial Intelligence
12	NC	CWRU	(12,4)	Computer Networks I
13	OS	RU STY1	(13,4)	Operating Systems
14	PD	NNSU-IPP	(14,13)	Introduction to Parallel Programming
15	PL	Utrecht-LC	(15,2)	Languages and Compilers
16	PL	215	(16,2)	Software Development Foundations
17	HCI	FIT3063	(17,16)	Human Computer Interaction
18	SE	CS169	(18,8)	Software Engineering
19	SF	CS2200	(19,12)	Computer Systems and Networks
20	SP	CMU-TEGS	-	Technology, Ethics, and Global Society
21	AL	CSC301	(21,8)	Analysis of Algorithms
22	PBD	CSWP	(22,3)	Web Platforms
23	PBD	CSMP	(24,3)	Mobile Platforms
24	AL	CSCI136	-	Data Structures and Advanced Programming

En 5.1 tenemos un ejemplo de plan de estudio y sus relaciones de un plan de estudios de la enseñanza de la informática, modelo conocido como el primer intento de ordenar el programa de estudios de la informática (ACM68). La difusión de ese estándar o modelo no tenía el impacto que permite en la actualidad la difusión de innovaciones o modelos internacionales, pero de alguna manera influenció el desarrollo de las propuestas educativas que han existido en el país. En 5.2 y 5.3 tenemos ejemplos de cursos modelos y sus relaciones según los referentes ACM2001 y ACM2013.

Tabla 5.4: Sistemas 2009 con Acm68

Id	Nombre	Relacion	IdUnificado	model1	model2
1	ma1	-	2	-	-
2	ca11	-	5	-	-
3	algo1	-	3	-	-
4	intro1	-	1	-	-
6	algo2	(6,3)	11	(11,3),(11,8)	(11,3)
7	ca12	(7,2)	9	(9,5),(9,4)	(9,5)
8	mat2	(8,1)	4	(4,2)	(4,2)
10	estruc1	(10,3)	10	(10,3),(10,6)	(10,3)
11	algo3	(11,8)	22	(22,6),(22, 17)	(22,10)
12	matd	(12,7)	6	(6,1)	(6,4)
13	e1	(13,6)	8	(8,2)	(8,9)
15	comg	(15,9)	21	(21,10)	(21,30)
17	ed	(17,6)	-	(14,9)	(14,9)
18	cd	(18,16)	-	(16,6),(16, 12)	(16,44)
19	e2	(19,13)	13	(13,8)	(13,8)
20	bd1	(20,22)	20	(20,10)	(20,45)
21	mn	(21,13)	7	(7,1),(7,5),(7,4)	(7,14)
22	lg	(22,9),(22,10)	18	(18,10)(18,11)	(18,22)(18,6)
23	dg	(23,14)	17	(17,6),(17, 16)	(17,16)
26	arq	(26,19)	12	(12,3),(12,8)	(12,17)
27	so	(27,18)	15	(15,10),(15,11),(15,12)	(15,18)
32	ia	-	24	(24,10)	(24,34)
33	mo	-	19	(19,15),(19,13)	(19,29)
34	net	-	23	(23,19)	(24,12)(24,15)
37	sd	-	25	(25,12)	(25,23)

Tabla 5.5: Sistemas 2017

Id	Nombre	Relacion	NombreCurso
1	ca1	calculo I	-
2	aal	algebra y geometria analítica	-
3	cal2	calculo II	-
4	fis1	física I	-
5	q1	química general	-
6	cal3	calculo III	-
7	pa1	Programación y fundamentos de algoritmica	-
8	ts	Teoría general de sistemas	-
9	e2	Estadística	-
10	sed	Series y ecuaciones diferenciales	(10,6)
11	md	Matemáticas Discretas	-
12	apo	Algoritmica y programación orientada a objetos	(12,7)
13	mn	Metodos numéricos	(13,9)
14	pm	Probabilidades y muestreo	(14,9)(14,10)
15	fis	Física electrónica y sistemas digitales	(15,10)
16	bd	Base de Datos	(16,12)
17	daa	Diseño y Análisis de Algoritmos	(17,12)
18	edd	Estructura de datos	(18,12)
19	asi	Analisis de Sistemas de Información	-
20	ms	Modelos y Simulación	(20,13)(20,14)
21	aq	Arquitectura de Computadoras	(21,15)
22	lc	Lenguajes y Compiladores	(22,11)
23	bid	Big Data	(23,16)(223,17)
24	cv	Computación Visual	-
25	ds	Diseño de Sistemas de Información	(25,19)
26	io	Investigación Operativa	(26,20)
27	rta	Redes Transmisión y Automatización y Control	(27,21)
28	so	Sistemas Operativos	-
29	ih	Interacción hombre computador	(29,24)
30	in	Inteligencia de Negocios	-
31	dsw	Desarrollo de Sistemas Web	(31,25)(31,23)
32	fwp	Formulación y Evaluación de Proyectos	(32,26)(32,25)
33	ia	Inteligencia Artificial	(33,26)(33,22)

En 5.4 y 5.5 tenemos una lista de cursos del programa de estudios de ingeniería de sistemas 2009 y 2017 respectivamente, con una columna que identifica el curso equivalente del modelo de ACM 68 en el primer caso y se hace lo mismo con los programas o referentes que queremos realizar la comparación, por motivos de espacio no incluiremos todas las tablas aquí, pero se incluyen como enlaces en el anexo del documento.

El proceso de modelamiento de la información de las curriculas pasa por establecer un $(ID, Nombre)$ que servirá para las comparaciones así como establecer dos modelos de grafos que usaremos para la comparación. Establecemos los siguientes pasos para el modelamiento:

1. Usamos una tabla de $(ID, Nombre)$ para cada guía o plan que queremos comparar.
2. Para realizar las comparaciones, establecemos un índice común que nos permita realizar la selección de elementos comunes.
3. Establecemos como $Modelo_1$ a la comparación que incluye los nodos y relaciones del grafo G_1 y se identifica la existencia de los mismos nodos y relaciones con el grafo G_2 . La codificación de las relaciones considera las dependencias del grafo G_1
4. Establecemos como $Modelo_2$ a la comparación que incluye los nodos y relaciones del grafo G_1 y se identifica la existencia de los mismos nodos y relaciones con el grafo G_2 . La codificación de las relaciones considera las dependencias del grafo G_1 y G_2 respectivamente. Es decir se comparan cada grafo considerando sus relaciones originales.

Para la comparación a nivel de cursos si requiere inferir para la codificación el alcance de las posibles unidades de conocimiento identificados en los planes de estudio de las universidades que estamos analizando. Dado el alcance en tiempo y revisión, solo mostraremos comparaciones de pocos cursos pero sentando las bases de la factibilidad según las hipótesis de la presente investigación.

Técnica analítica

Para el análisis de identificación de relaciones usaremos la teoría de grafos modelando los planes, guías internacionales y cursos siguiendo una representación de grafos. Posteriormente usaremos propiedades de la teoría de grafos para encontrar similitud, distancias, sub-grafos comunes y valores cuantitativos que indiquen la cantidad de relaciones comunes entre ambos modelos a comparar. Como se explica en 2.4 usaremos el modelamiento de grafos y los algoritmos respectivos para identificar similitud y valores cuantitativos.

Nuestra hipótesis central es la siguiente:

1. P1. (H1) Dado un modelo en grafos de un plan de estudios o una guía internacional encontrar la relación entre un plan de estudios G_1 y G_2 como la métrica entre ambos modelos usando el concepto de similitud grafos. De modo que la similitud es significativa.

Dada la métrica que consideramos en el estudio:

$$\delta(g, g') = 1 - \frac{|mcs(g, g')|}{\max(|g|, |g'|)}$$

Que a su vez depende de encontrar el Máximo común sub-grafo MCS que en el presente estudio depende de un algoritmo indicado en 2.5. Con los datos indicados previamente modelaremos planes, guías internacionales cursos como grafos y encontraremos la métrica para validar las respectivas hipótesis propuestas.

Las hipótesis secundarias se enumeran a continuación:

1. P2. (H2) Dado un grafo G_1 que representa una profesión de ingeniería de sistemas o informática y G_2 un referente internacional de ACM/IEEE u otro. Identificar que la similitud es significativa.
2. P3. (H3) Dado un grafo G_1 que representa el plan de estudio de una carrera vinculada a la informática de un año X y G_2 un plan de la misma carrera pero del año Y . Identificar que la similitud es significativa.

Como se indico en el capitulo 3 tendremos 2 métricas para evaluar nuestras hipótesis.

El nivel de similaridad entre 2 grafos como el valor $z = 0.2$ como el valor que indique o no la presencia significativa de similaridad. Un valor mayor a z refleja un nivel aceptable de similaridad, un valor menor indica que no existe similaridad.

Otro elemento que podemos considerar es la proporción en una de los planes o guías respecto a la cantidad de nodos en común. En este caso usaremos una proporción $t = 0.1$ para indicar cierto grado de similaridad por nodos. Esto significa que un valor menor a t significa que no existe similaridad en cantidad de nodos en común.

5.2 Pruebas de hipótesis

Tal como planteamos en el capitulo anterior de metodología, se sigue una serie de pasos para codificar los datos en grafos. En la siguiente tabla se muestra los datos recolectados de los planes y guías consideradas para el experimento.

Tabla 5.6: Tabulación de datos sobre comparaciones

Caso 1	Modelo model1	G(acm68)-n 30	G(acm68)-e 39	H(sis2009)-n 41	H(sis2009)-e 34
1	model2	30	39	42	32
Caso 2	Modelo model1	G(acm68)-n 30	G(acm68)-e 39	H(sis2017)-n 38	H(sis2017)-e 25
2	model2	30	39	38	24
Caso 3	Modelo model1	G(sis2009)-n 30	G(sis2009)-e 39	H(sis2017)-n 38	H(sis2017)-e 25
3	model2	30	39	38	24
Caso 4	Modelo model1	G(acm2013)-n 30	G(acm2013)-e 39	H(sis2017)-n 38	H(sis2017)-e 25
4	model2	30	39	38	24
Caso 5	Modelo model1	G(acm2013)-n 26	G(acm2013)-e 20	H(pucp2020)-n 60	H(pucp2020)-e 20
5	model2	26	20	60	17
Caso 6	Modelo model1	G(acm68)-n 30	G(acm68)-e 39	H(unisis2018)-n 27	H(unisis2018)-e 15
6	model2	30	39	27	18

5.3 Presentación de Resultados

Presentaremos ahora los resultados organizados por casos. Es importante explicar algunas características de las visualizaciones:

1. Por cada caso se dan dos comparaciones, denominadas *Modelo₁* y *Modelo₂*.
2. El *Modelo₁* representa la comparación de los nodos y enlaces del grafo G_1 y la busquedad de similitud en el grafo G_2 . Las relaciones se consideran priorizando las relaciones del grafo G_1
3. El *Modelo₂* representa la comparación de los nodos y enlaces del grafo G_1 y la busquedad de similitud en el grafo G_2 . Las relaciones se consideran respetando las relaciones del grafo G_1 y las relaciones del grafo G_2 respectivamente.
4. Los enlaces comunes a ambos grafos aparecen en color naranja.
5. Los enlaces aparecen en color azul.
6. Los nodos del modelo en amarillo.
7. Los nodos del MCS en color verde

Caso 1 Hipotesis Secundaria 1

A continuación se muestran las representaciones gráficas y los resultados del *Caso₁* de la comparación entre un plan de estudio: Sistemas 2009 y la guía internacional ACM 68. Como podemos observar en la figura 5.1 y la figura 5.2 se muestra la identificación de un maximo común sub-grafo. En el primer ejemplo se muestra la comparación de un modelo de curricula de ingeniería de sistema del 2009 codificado y comparado con un modelo clásico de curricula de computación de acm68. Como se ha indicado previamente la codificación de las relaciones fuerza de alguna manera a la existencia de un sub-grafo teniendo en cuenta que el modelo 1 representa la identificación de nodos comunes y toma los enlaces considerando las relaciones del modelo G en este caso del ACM 68. En en el caso del modelo 2 se respetan las relaciones del

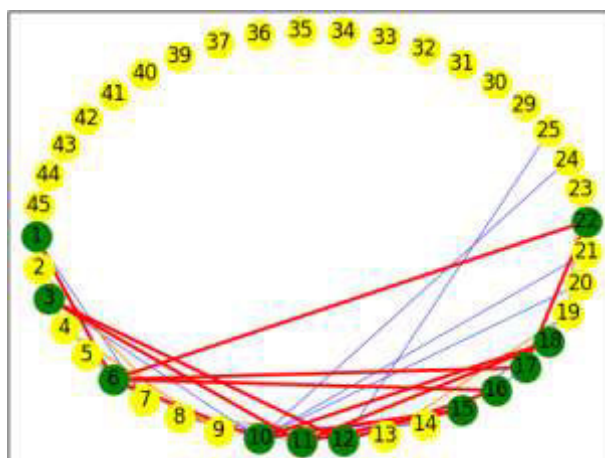


Figura 5.1: Sis2009 ACM68 MCS modelo 1

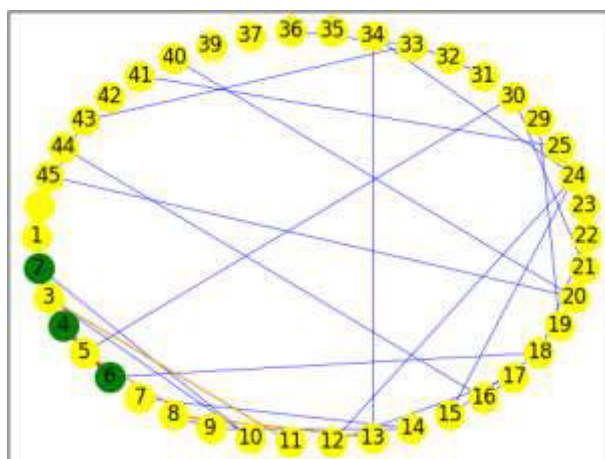


Figura 5.2: Sis2009 ACM68 MCS modelo 2

Tabla 5.7: Tabulación de datos sobre comparaciones

Caso	Modelo	G(acm68)-n	G(acm68)-e	H(sis2009)-n	H(sis2009)-e
1	model1	30	39	41	34
1	model2	30	39	42	32

Tabla 5.8: Comparación G-ACM 68 y H-Sistemas 2009

Caso	Modelo	MCSsubgrado	similaridad	nodos	enlaces
1	<i>modelo₁</i>	0.64285714	$z = 0.35714286$	11	13
1	<i>modelo₂</i>	0.9285714286	$z = 0.07142857143$	3	2

Tabla 5.9: Tabla de resultados de proporcionalidad

Caso	Modelo	nodosCom	enlacesCom	proporcionG	proporcionH
1	<i>modelo₁</i>	20	21	$t_1 = 0.6666666667$	$t_2 = 0.487804878$
1	<i>modelo₂</i>	9	5	$t_1 = 0.3$	$t_2 = 0.2142857143$

Tabla 5.10: Tabla de Hipotesis vs Resultados

Hipotesis	Caso	modelo	Similaridad	ProporcionG	ProporcionH
P2	1	modelo1	$z > 0.2$	$t_1 > 0.1$	$t_2 > 0.1$
P2	1	modelo2	$z < 0.1$	$t_1 > 0.1$	$t_2 < 0.1$

Caso 2 Hipotesis Secundaria 1

A continuación se muestran las representaciones gráficas y los resultados del *Caso 2* de la comparación entre un plan de estudio: Sistemas 2017 y la guía internacional ACM 68. Las comparaciones muestran en la figura 5.3 la comparación asumiendo que los nodos respetan la red de relaciones de acm68 y en la figura 5.4 se muestra el grafo considerando las relaciones

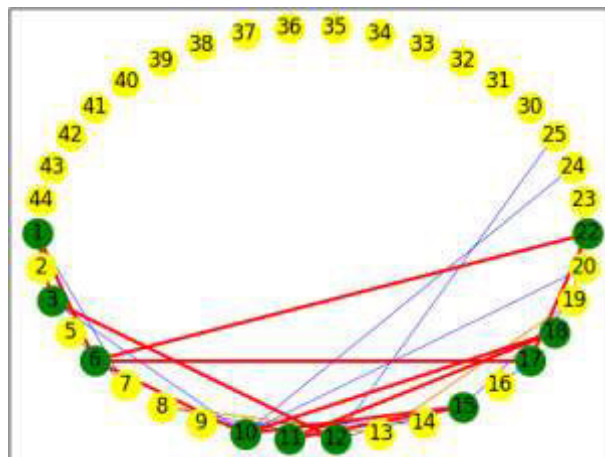


Figura 5.3: Sis2017 ACM68 MCS modelo 1

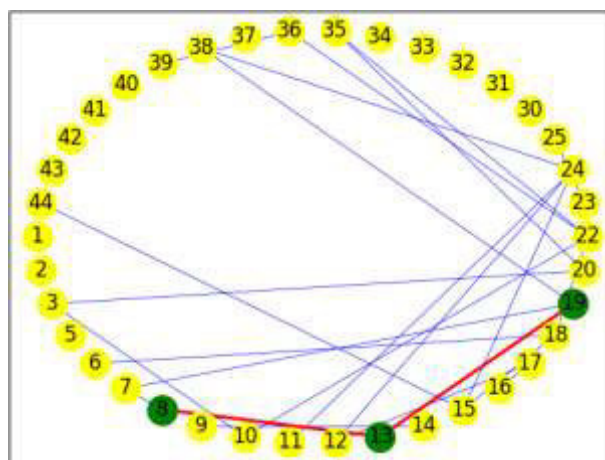


Figura 5.4: Sis2017 ACM68 MCS modelo 2

Tabla 5.11: Tabulación de datos sobre comparaciones

Caso	Modelo	G(acm68)-n	G(acm68)-e	H(sis2009)-n	H(sis2009)-e
2	model1	30	39	38	25
2	model2	30	39	38	24

Tabla 5.12: Comparación G-ACM 68 y H-Sistemas 2009

Caso	Modelo	MCSsubgrado	similaridad	nodos	enlaces
2	$modelo_1$	0.736842105263158	$z = 0.2631578947$	10	11
2	$modelo_2$	0.9210526316	$z = 0.07894736842$	3	2

Tabla 5.13: Tabla de resultados de proporcionalidad

Caso	Modelo	nodosCom	enlacesCom	proporcionG	proporcionH
2	$modelo_1$	14	14	$t_1 = 0.4666666667$	$t_2 = 0.3684210526$
2	$modelo_2$	3	2	$t_1 = 0.1$	$t_2 = 0.07894736842$

Tabla 5.14: Tabla de Hipotesis vs Resultados

Hipotesis	Caso	modelo	Similaridad	ProporcionG	ProporcionH
P2	2	modelo1	$z > 0.2$	$t_1 > 0.1$	$t_2 > 0.1$
P2	2	modelo1	$z < 0.2$	$t_1 > 0.1$	$t_2 > 0.1$

Caso 3 Hipotesis Secundaria 2

A continuación se muestran las representaciones gráficas y los resultados del *Caso 3* de la comparación entre un plan de estudio: Ingeniería de Sistemas 2009 e Ingeniería de Sistemas 2017. Este caso se representa en la figura 5.5 y en la figura 5.6. En la primera imagen se muestra una identificación del valor de similaridad teniendo como base al modelo de grafos del plan 2009.

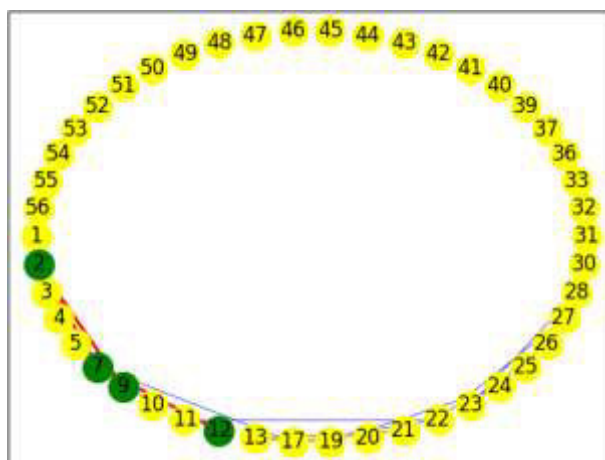


Figura 5.5: Sis2009 Sis2017 MCS modelo 1

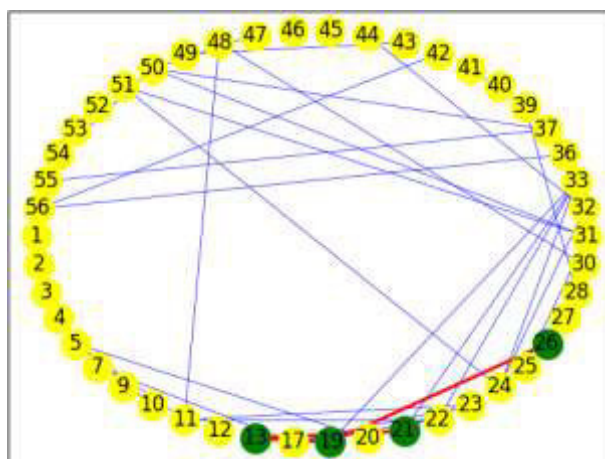


Figura 5.6: Sis2009 Sis2017 MCS modelo 2

Tabla 5.15: Tabulación de datos sobre comparaciones

3	model1	30	39	38	25
3	model2	30	39	38	24

Tabla 5.16: MCS

Caso	Modelo	MCSsubgrado	similaridad	nodos	enlaces
3	<i>modelo</i> ₁	0.914893617	$z = 0.08510638298$	4	3
3	<i>modelo</i> ₂	0.914893617	$z = 0.08510638298$	4	3

Tabla 5.17: Tabla de resultados de proporcionalidad

Caso	Modelo	nodosCom	enlacesCom	proporcionG	proporcionH
3	<i>modelo</i> ₁	7	5	$t_1 = 0.1489361702$	$t_2 = 0.152173913$
3	<i>modelo</i> ₂	4	3	$t_1 = 0.08510638298$	$t_2 = 0.08695652174$

Tabla 5.18: Tabla de Hipotesis vs Resultados

Hipotesis	Caso	modelo	Similaridad	ProporcionG	ProporcionH
P3	3	modelo1	$z < 0.2$	$t_1 > 0.1$	$t_2 > 0.1$
P3	3	modelo1	$z < 0.2$	$t_1 < 0.1$	$t_2 < 0.1$

Caso 4 Hipotesis Secundaria 2

A continuación se muestran las representaciones gráficas y los resultados del *Caso 4* de la comparación entre un plan de estudio: Ingeniería de sistemas 2017 y ACM 2013. En la figura 5.7 se representa la comparación entre grafos priorizando los enlaces de ACM 2013 y en la figura 5.8 cada grafo mantiene sus relaciones.

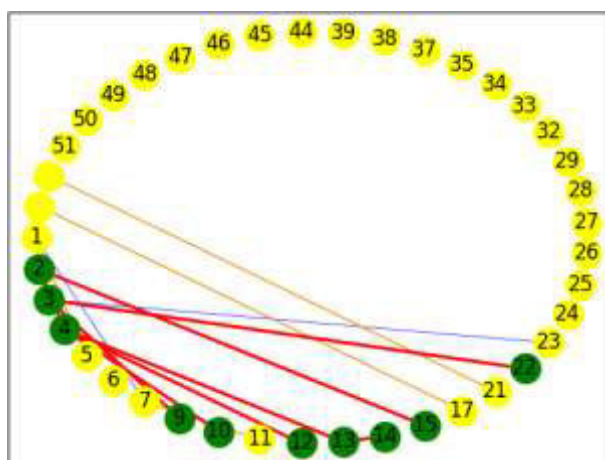


Figura 5.7: Sis2017 y ACM 2013 MCS modelo 1

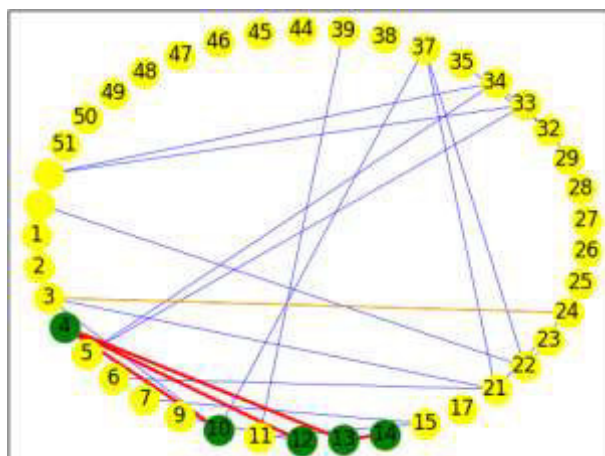


Figura 5.8: Sis2017 y ACM 2013 MCS modelo 2

Tabla 5.19: Tabulación de datos sobre comparaciones

4	model1	30	39	38	25
4	model2	30	39	38	24

Tabla 5.20: MCS

Caso	Modelo	MCSsubgrado	similaridad	nodos	enlaces
4	$modelo_1$	0.756097561	$z = 0.243902439$	10	9
4	$modelo_2$	0.8780487805	$z = 0.1219512195$	5	4

Tabla 5.21: Tabla de resultados de proporcionalidad

Caso	Modelo	nodosCom	enlacesCom	proporcionG	proporcionH
4	$Modelo_1$	18	13	$t_1 = 0.75$	$t_2 = 0.4390243902$
4	$Modelo_2$	8	6	$t_1 = 0.3333333333$	$t_2 = 0.1951219512$

Tabla 5.22: Tabla de Hipotesis vs Resultados

Hipotesis	Caso	modelo	Similaridad	ProporcionG	ProporcionH
P2	4	modelo1	$z > 0.02$	$t_1 > 0.1$	$t_2 > 0.1$
P2	4	modelo1	$z < 0.02$	$t_1 > 0.1$	$t_2 > 0.1$

Caso 5 Hipotesis Secundaria 2

A continuación se muestran las representaciones gráficas y los resultados del *Casos* de la comparación entre un plan de estudio de ingeniería informática y una guía internacional ACM 2013. En la comparación del modelo 1, se representa con la figura 5.9 con grafos y enlaces del modelo ACM y en la figura 5.10 que representa cada grafo con sus respectivos enlaces.

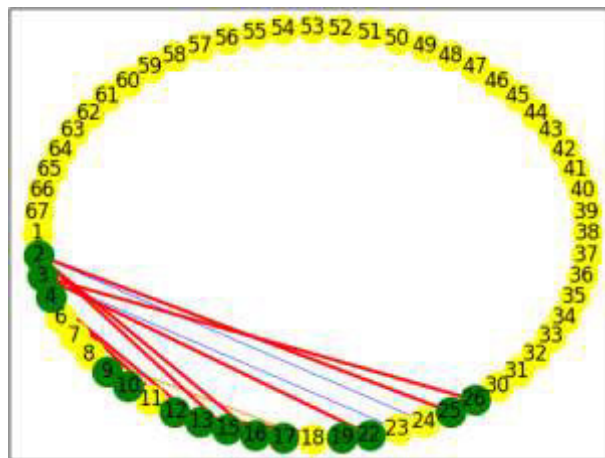


Figura 5.9: IngInform2020 y ACM 2013 MCS modelo 1

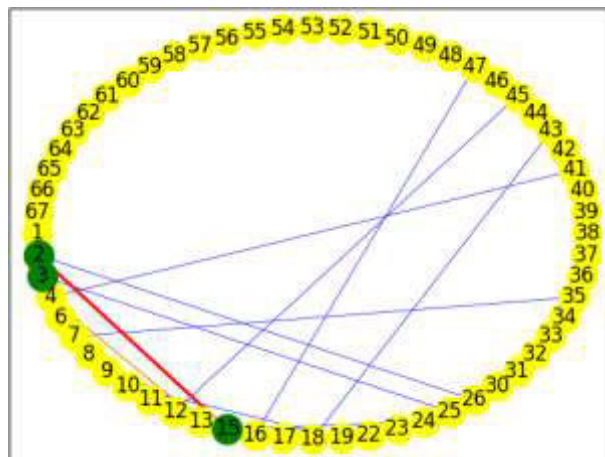


Figura 5.10: IngInform2020 y ACM 2013 MCS modelo 2

Tabla 5.23: Tabulación de datos sobre comparaciones

5	model1	26	20	60	20
5	model2	26	20	60	17

Tabla 5.24: MCS

Caso	Modelo	MCSsubgrado	similaridad	nodos	enlaces
5	$modelo_1$	0.7666666666666666	$z = 0.2333333333$	18	16
5	$modelo_2$	0.95	$z = 0.05$	5	3

Tabla 5.25: Tabla de resultados de proporcionalidad

Caso	Modelo	nodosCom	enlacesCom	proporcionG	proporcionH
5	$Modelo_1$	18	16	$t_1 = 0.6923076923$	$t_2 = 0.3$
5	$Modelo_2$	5	3	$t_1 = 0.1923076923$	$t_2 = 0.0833333333$

Tabla 5.26: Tabla de Hipotesis vs Resultados

Hipotesis	Caso	modelo	Similaridad	ProporcionG	ProporcionH
P2	5	modelo1	$z > 0.02$	$t_1 > 0.1$	$t_2 > 0.1$
P2	5	modelo1	$z < 0.02$	$t_1 > 0.1$	$t_2 < 0.1$

Caso 6 Hipotesis Secundaria 1

A continuación se muestran las representaciones gráficas y los resultados del *Caso 6* de la comparación entre un plan de estudio de ingeniería de sistemas(UNI) y una guía internacional ACM 68. En la comparación del modelo 1, se representa con la figura 5.11 con grafos y enlaces del modelo ACM y en la figura 5.12 que representa cada grafo con sus respectivos enlaces.

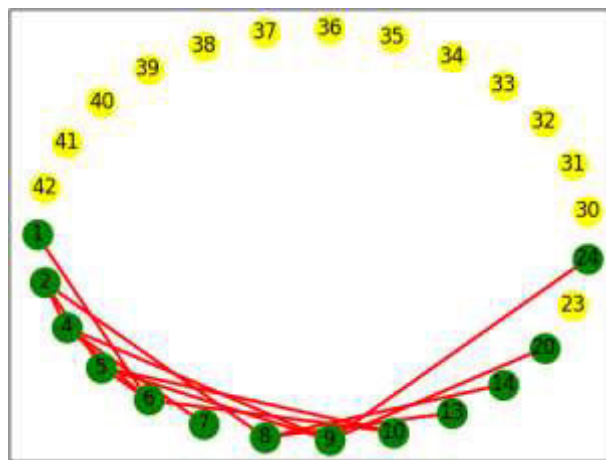


Figura 5.11: IngSistemas2018 y ACM 68 MCS modelo 1

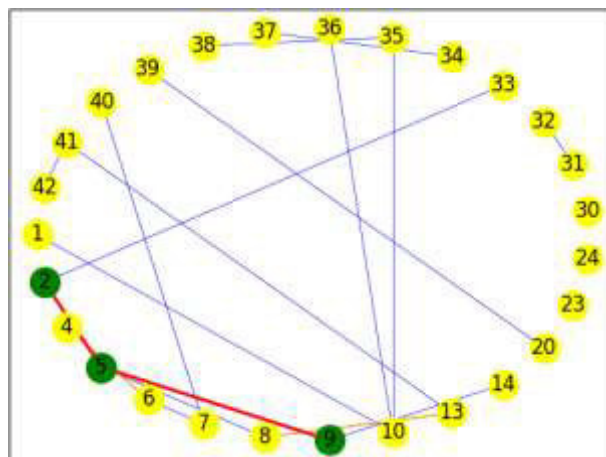


Figura 5.12: IngSistemas2018 y ACM 68 MCS modelo 2

Tabla 5.27: Tabulación de datos sobre comparaciones

6	model1	30	39	27	15
6	model2	30	39	27	18

Tabla 5.28: MCS

Caso	Modelo	MCSsubgrado	similaridad	nodos	enlaces
6	<i>modelo₁</i>	0.566666666666667	$z = 0.4333333333$	13	15
6	<i>modelo₂</i>	0.9	$z = 0.1$	3	2

Tabla 5.29: Tabla de resultados de proporcionalidad

Caso	Modelo	nodosCom	enlacesCom	proporcionG	proporcionH
6	<i>Modelo₁</i>	13	15	$t_1 = 0.43$	$t_2 = 0.3$
6	<i>Modelo₂</i>	7	4	$t_1 = 0.1$	$t_2 = 0.0833333333$

Tabla 5.30: Tabla de Hipotesis vs Resultados

Hipotesis	Caso	modelo	Similaridad	ProporcionG	ProporcionH
P1	6	modelo1	$z > 0.02$	$t_1 > 0.1$	$t_2 > 0.1$
P1	6	modelo2	$z < 0.02$	$t_1 < 0.1$	$t_2 < 0.1$

5.4 Resultados

Como podemos observar no en todos los casos se logra obtener una similaridad mínima es decir cuando $z > 0.2$ lo que invalida el caso de la hipótesis considerada en el estudio. El uso o no de las relaciones originales en la equivalencia de cursos. Es decir cuando usamos las relaciones originales de las curriculas internacionales se encuentra más casos de similaridad, pero también se encuentran casos que teniendo las relaciones del *Modelo₂* se da un grado de similaridad. Esto es importante subrayar pues el modelamiento y la comparación nos puede dar pistas sobre que áreas y que cursos servirán como elementos para una incorporación o adaptación a perfiles de una curricula o guía internacional sin que eso afecte la estructura orgánica de la propuesta educativa.

En el caso del indicador $t > 0.1$ podemos observar el cumplimiento de los casos de proporcionalidad con mayor incidencia en el *Modelo₁* eso nos da una pista que en algunos casos las relaciones de dependencia de las guías internacionales pueden

acercar más la adopción de los referentes en contraste a que las instituciones usan sus propios esquemas de dependencia entre cursos, es decir no solo es necesario incorporar cursos y áreas de conocimiento sino también observar las dependencias entre estos cursos o contenidos.

5.5 Conclusiones

1. El modelamiento en grafos para la información de planes de estudio como de guías internacionales y habilitar la opción de comparación con indicadores numéricos es una herramienta novedosa para abordar el proceso de innovación curricular de las instituciones.
2. Más allá de las diferencias estructurales o históricas de las escuelas peruanas, utilizando herramientas como las propuestas en el presente estudio una institución puede observar el cambio en el tiempo y la adaptación a los distintos enfoques que va teniendo del diseño curricular. Además que puede observar si esa orientación refleja algún perfil o contenido central. El alcance de esta investigación no ha abordado la comparación en concreto de niveles de profundidad o tipos de orientación de los cursos, dado que eso implicaría un nivel más elaborado de comparaciones. Pero si se plantea el método y los mecanismos que pueden permitir ese nivel de evaluación de esos casos.
3. El plantear comparaciones entre planes de estudio de la diversidad de oferta educativa así como encontrar similitudes con guías internacionales apoya la inclusión de los referentes internacionales así como puntos de partida para las instituciones.
4. La inclusión de un modelo de referente internacional como el ACM68 y compararlo con al menos un plan de estudio antiguo de una universidad nos sirve para encontrar pistas de la influencia de la computación internacional en décadas pasadas y en parte también dar pistas que los planes de estudio pueden seguir la línea de profesiones de informática internacional así lleven nombres de profesiones de ingeniería.

5.6 Trabajos futuros

El mundo de la educación en informática es cambiante como la disciplina misma, mientras se le observa se va adaptando y cambiando en el tiempo. Sin embargo lo que menos cambia es la institucionalidad y eso se refleja tanto en los diseños orgánicos de las instituciones educativas como de los planes de oferta educativa, las personas y los elementos culturales. Innovar en estos sistemas sociales es un verdadero desafío que solo tiene sentido con el impulso de cada organización. Este estudio se presenta como una oportunidad para contribuir con herramientas de análisis cuantitativo en el estudio del desarrollo de los cambios y de la arquitectura interna de las relaciones de los contenidos y cursos de los programas educativos. Es un factor que no deberíamos ignorar como un elemento que nos permita contribuir en el desarrollo de los diseños curriculares y observar los movimientos y relaciones que puedan existir en los programas académicos.

Como trabajos futuros quedan poder hacer comparaciones a nivel de competencias de los planes de estudio y sus correspondientes guías internacionales o pares educativos. También poder hacer comparaciones a nivel de propiedades de los grafos, como por ejemplo el nivel de orientación teórico práctico de una grafo que corresponda a una unidad de conocimiento o curso. Poder realizar análisis de concentraciones o de agrupamientos de áreas también puede mejorar la toma de decisiones en la planificación educativa y vincular eso a otras necesidades económicas como industriales.

En cierta forma el modelamiento de los grafos y sus relaciones requiere juicio de análisis y por ende corre riesgo de introducir errores de interpretación, sin embargo esto puede reducirse conforme se puedan lograr consensos e interpretaciones por contenidos de los cursos. Es decir aplicando técnicas estadísticas y umbrales de aproximación en los contenidos de los cursos se podría lograr niveles de equivalencias entre nodos luego de las relaciones y poder ejecutar comparaciones más cercanas o automatizadas, introducir una capacidad de generalidad y escalamiento del proceso.

Finalmente esta línea de investigación abre posibilidades para generar mejoras en la educación universitaria desde la comunidad académica de la informática en nuestras universidades.

BIBLIOGRAFÍA

- (1991). *Computing Curricula 1991: Report of the ACM/IEEE-CS Joint Curriculum Task Force*. Technical report, New York, NY, USA.
- Alvarado, J. (2020). Data para comparación. <https://docs.google.com/spreadsheets/d/17WweMjL7o8wkOY0YJ-GygTL1D3N6cfEy9gB1nZQWla4/edit?usp=sharing>.
- Alvarado-Revatta, J. (2015). Adoption alternatives of academic innovations in computer science schools in peru. In *2015 Latin American Computing Conference (CLEI)* (pp. 1–12).
- Alvarez, M. a., Baiocchi, J., & Pow Sang, J. A. (2008). Computing and higher education in Peru.
- ANECA (2004). *Libro Blanco: Título de grado en ingeniería informática*.
- Atchison, W. F., Conte, S. D., Hamblen, J. W., Hull, T. E., Keenan, T. A., Kehl, W. B., McCluskey, E. J., Navarro, S. O., Rheinboldt, W. C., Schweppe, E. J., Viavant, W., & Young, D. M. (1968). Curriculum 68: Recommendations for academic programs in computer science: A report of the acm curriculum committee on computer science. *Commun. ACM*, 11(3), 151–197.
- Bauer, F. L. (2010). *Origins and Foundations of Computing*. Springer, Berlin, Heidelberg.
- Britannica (2020). Enciclopedia en línea británica.
- Bunke, H. (2000). Graph matching: Theoretical foundations, algorithms, and applications. In *In Proceedings of Vision Interface 2000, Montreal* (pp. 82–88).
- CC, U. (2019). Cincuenta años de historia del centro de cómputo uni.
- CIP (2006). Denominaciones Y Perfiles De Las Carreras En Ingeniería De Sistemas, Computación E Informática. <http://sis-com-inf.pucp.pe/informe2006>. Accessed: 2020-10-27.
- Contreras Véliz, J. L., Alanoca Gutiérrez, J., Salem Barbar, J., Quevedo Reyes, J. E., Garita, G., Sepúlveda Lima, R., Hinojosa Raza, C., Duarte Pavón, H. J., Chávez Cervantes, A. P., Estrada Quintero, A. E., Bernal, D., García, M. E., Pow-Sang, J. A., & González, L. (2013). *Tuning LA: Educación Superior en América Latina: reflexiones y perspectivas en Informática*. Technical report.
- Denning, P. J. (2003). *Computer Science*, (pp. 405–419). John Wiley and Sons Ltd.: GBR.
- Denning, P. J. (2005). Is computer science science? *Commun. ACM*, 48(4), 27–31.
- Denning, P. J. & Riehle, R. D. (2009). The profession of it: Is software engineering engineering? *Commun. ACM*, 52(3), 24–26.

- Fein, L. (1959). The role of the university in computers, data processing, and related fields. *Commun. ACM*, 2(9), 7–14.
- Forsythe, G. E. (1968). Computer science and education. In A. J. H. Morrel (Ed.), *Information Processing, Proceedings of IFIP Congress 1968, Edinburgh, UK, 5-10 August 1968, Volume 2 - Hardware, Applications* (pp. 1025–1039).
- Gao, X., Xiao, B., Tao, D., & Li, X. (2010). A survey of graph edit distance. *Pattern Analysis and Applications*, 13(1), 113–129.
- Gonzales de Olarte, E. (1990). Una economía bajo violencia: peru, 1980-1990.
- Horst Bunke, S. K. (1998). A graph distance metric based on the maximal common subgraph. In *Pattern Recognition Letters* (pp. 255–259).
- INCOSE (2011). INCOSE Systems Engineering Handbook v. 3.2.2. *SE Handbook Working Group*, (October), 1306.
- Joint Task Force on Computing Curricula, A. f. C. M. A. & Society, I. C. (2013). *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. New York, NY, USA: Association for Computing Machinery.
- Knuth, D. E. (1972). George forsythe and the development of computer science. *Commun. ACM*, 15(8), 721–726.
- Koutra, D., Ramdas, A., Parikh, A., & Xiang, J. (2011). Algorithms for graph similarity and subgraph matching.
- Lightfoot, J. M. (2010). A graph-theoretic approach to improved curriculum structure and assessment placement. In *Communications of the IIMA*.
- Magin (2019). maximum common subgraph in a directed-graph. <https://stackoverflow.com/questions/43108481/maximum-common-subgraph-in-a-directed-graph>.
- Marshall, L. (2012). A comparison of the core aspects of the acm/ieee computer science curriculum 2013 strawman report with the specified core of cc2001 and cs2008 review. In *Proceedings of Second Computer Science Education Research Conference, CSERC '12* (pp. 29–34). New York, NY, USA: Association for Computing Machinery.
- Mclauchlan de Arregui, P. (1990). Educación superior en el Perú: Datos para el análisis. *GRADE*.
- Mulder, F. & van Weert, T. (2001). Ifip/unesco's informatics curriculum framework 2000 for higher education. *SIGCSE Bull.*, 33(4), 75–83.
- NetWorkX (2022).
- Ortiz, B. (2020). Gisella orjeda: No hay que asociar la ciencia solo con la salud | entrevista.
- Pasterk, S. & Bollin, A. (2017). Graph-based analysis of computer science curricula for primary education. In *2017 IEEE Frontiers in Education Conference (FIE)* (pp. 1–9).

- Perekatov, V. (2008). COMPUTER SCIENCE STANDARDS IN HIGHER SCHOOL OF THE U . S . (1968 – 2001). (pp. 1–25).
- project, T. (2006). *Una introducción a Tuning Educational Structures in Europe*. Technical report.
- Pucp (2017). 25 años de ingeniería informática.
- Ramos, T. J., Micheloud, O. M., Painter, R., & Kam, M. (2013). Nomenclatura recomendada por el IEEE para programas Universitarios del área computacional en latinoamerica.
- Rivera, L., Piedra, J., Mauricio, D., Luza, C., Guerra, J., & De la Cruz, P. (2008). Carreras de Ciencia y Tecnología de Computación en San Marcos.
- Rogers, E., Morgenthaler, J., & Morgenthaler, F. (1962). *Diffusion of Innovations*. Free Press of Glencoe.
- Sabin, M., Viola, B., Impagliazzo, J., Angles, R., Curiel, M., Leger, P., Murillo, J., Nina, H., Pow-Sang, J. A., & Trejos, I. (2016). Latin american perspectives to internationalize undergraduate information technology education. In *Proceedings of the 2016 ITiCSE Working Group Reports*, ITiCSE '16 (pp. 1–22). New York, NY, USA: Association for Computing Machinery.
- Sedgewick, R. & Wayne, K. (2011). *Algorithms*. Addison-Wesley Professional, 4th edition.
- Shackelford, R., McGettrick, A., Sloan, R., Topi, H., Davies, G., Kamali, R., Cross, J., Impagliazzo, J., LeBlanc, R., & Lunt, B. (2006). Computing curricula 2005: The overview report. *SIGCSE Bull.*, 38(1), 456–457.
- The Joint Task Force on Computing Curricula, C. (2001). Computing curricula 2001. *J. Educ. Resour. Comput.*, 1(3es), 1–es.
- Tucker, A. B. (1991). Computing curricula 1991. *Commun. ACM*, 34(6), 68–84.
- Ullmann, J. R. (1976). An algorithm for subgraph isomorphism. *JOURNAL OF THE ACM*, 28(1), 31–42.
- Yamada, G., Lavado, P., & Martinez, J. J. (2014). ¿Una promesa incumplida? La calidad de la educación superior universitaria y el subempleo profesional en el Perú.
- Zakimi, B. & Reátegui, N. (2012). *Percepción en gerentes y docentes de la pertinencia de incorporación de competencias profesionales de ingenieros informáticos en el Perú*. PhD thesis, Universidad San Ignacio de Loyola.

Apéndice A

ANEXOS

A.1 Documento de codificación de datos

En el presente documento se presenta las codificaciones de los cursos y relaciones de guías internacionales así como de planes de estudio de universidades que se han considerado en el ejercicio de análisis del proyecto.

Dada las características de hoja de cálculo se incluye el enlace permanente para su acceso: enlace. Se ha incluido también una referencia en la bibliografía en: (Alvarado, 2020)

A.2 Repositorio de imágenes y datos codificados

En el siguiente repositorio git se encuentran las imágenes y los archivos CSV que tienen la codificación utilizada para las comparaciones presentadas en el estudio.

enlace o acceder al siguiente url: <https://github.com/jorge-alvarado-revata/comparatool>