



Применение графовых моделей в проектном управлении

© 2023, А.З. Асанов¹✉, И.Ю. Мышкина², Л.Ю. Грудцына²

¹ Институт искусственного интеллекта МИРЭА – Российского технологического университета, Москва, Россия

² Набережночелнинский институт (филиал) Казанского (Приволжского) федерального университета, Набережные Челны, Россия

Аннотация

Исследуется возможность применения графовых моделей в проектном управлении для решения задачи формирования проектной группы и распределения работ проекта между потенциальными исполнителями. Выделены возможные направления применения онтологического моделирования в проектном управлении. Средствами редактора онтологий *Protégé* реализована онтология проекта. Исследуется возможность построения векторных представлений элементов простых графов и графов знаний в задаче формирования проектной группы. Задачи исследования: построение графовых моделей проектов – простого графа и графа знаний; применение векторных представлений вершин и отношений графовых моделей для решения задачи распределения работ проекта. Рассматривается возможность применения мер семантической близости в онтологии. Новизна предложенного подхода заключается в использовании при формировании проектной группы данных об участниках реализованных проектов, что повышает вероятность успешной реализации проекта. Представление элементов графа в виде числовых векторов позволяет применить к исследованию предметной области методы машинного обучения. Приводится пример решения задачи формирования рабочей группы проекта, в котором осуществлён выбор исполнителей проектной работы. Все расчёты осуществлены средствами языка *Python*.

Ключевые слова: графовая модель, граф знаний, онтология, проект, команда, исполнитель.

Цитирование: Асанов А.З., Мышкина И.Ю., Грудцына Л.Ю. Применение графовых моделей в проектном управлении // Онтология проектирования. 2023. Т.13, №2(48). С.232-242. DOI:10.18287/2223-9537-2023-13-2-232-242.

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.

Введение

Задачи управления кадрами организации по степени важности стоят в одном ряду с задачами производства и сбыта продукции. Всё больше компаний становятся проектно-ориентированными, выполняя работы в рамках проектов [1]. Успешность нового проекта в любой сфере деятельности напрямую зависит от компетентности выбранных исполнителей, от того, правильно ли распределены между членами команды отдельные задачи проекта [2]. В данной работе рассматривается задача распределения работ проекта между потенциальными исполнителями и исследуется возможность применения для её решения графовых моделей (ГМ). Использование ГМ позволяет построить структурную схему исследуемой предметной области (ПрО), наглядно и понятно представить то, как связаны и взаимодействуют её объекты. Использование современного математического аппарата при анализе ГМ позволяет устанавливать ранее не известные связи, зависимости и закономерности, отслеживать изменения различных характеристик. Особая ценность ГМ заключается в том, что они

предоставляют формальный аппарат для плохоформализуемых задач, требующих учёта как количественных, так и качественных параметров, в т.ч. в условиях неопределённости. В последнее время для решения прикладных задач используют такие ГМ, как байесовские сети, марковские цепи, когнитивные модели (карты), искусственные нейронные сети, графы знаний (ГЗ), онтологии и др. Вероятностные ГМ используются в компьютерном зрении, биоинформатике, принятии решений, робототехнике и др. [3]. Построение ГМ, как правило, осуществляется путём привлечения экспертов. В этом случае ГМ есть субъективное представление эксперта о ПрО в виде схемы взаимодействия её объектов. Кроме привлечения экспертов, ГМ могут выводиться из данных на основе методов машинного обучения (МО) [3].

В данной работе исследуется возможность применения ГМ и ГЗ к решению одной из задач проектного управления.

1 Онтологическая модель проекта

Далее приведены определения ГМ, используемые в работе.

Обобщённый граф или просто *граф* G состоит из непустого конечного множества $V(G)$ элементов, называемых вершинами, и конечного семейства $E(G)$ пар (не обязательно различных) элементов из $V(G)$, именуемых рёбрами; использование слова «семейство» допускает существование кратных рёбер. Если пары вершин семейства $E(G)$ не упорядочены, то граф называется неориентированным, если упорядочены – ориентированным графом (орграфом). В *простом графе* имеется не более одного ребра, соединяющего пару вершин, а также отсутствуют петли (рёбра, соединяющие вершину с самой собой) [4].

ГЗ – это орграф, который может содержать кратные рёбра и петли, каждое из рёбер относится к одному из возможных типов; каждому заданному типу ребра соответствует некоторое отношение между объектами ПрО, а вершинам – объектам ПрО.

Онтология – это формальная, явная спецификация общей концептуализации, состоящая из иерархии понятий ПрО, связей между ними и ограничений, которые действуют в рамках этой модели. Термин «онтология» связывают с термином «ГЗ», поскольку при наполнении онтологии экземплярами происходит её превращение в ГЗ [5].

Онтология есть кортеж $\langle L := L^C \cup L^P \cup L^A \cup L^{VA}, C, P, A, F, G, H^C, J, H^P, I \rangle$, где L – лексикон, т.е. множество терминов понятий (L^C), отношений (L^P), атрибутов (L^A), значений атрибутов (L^{VA}); C – множество понятий; $P = C \times C$ – множество отношений между понятиями; $A = C \times L^{VA}$ – множество атрибутов понятий; $F = L^C \rightarrow C$ – функция связи лексикона с понятиями; $G = L^P \rightarrow P$ – функция связи лексикона с отношениями; $J = L^A \rightarrow A$ – функция связи лексикона с атрибутами; $H^C = C \times C$ – таксономическая иерархия классов; $H^P = P \times P$ – иерархия отношений, I – множество экземпляров (понятий единичного объёма) [6].

Применение онтологического подхода позволяет [7, 8]: совместное использование пользователями знаний в рассматриваемой ПрО; установление общей терминологии, определение её смысла и причинно-следственных связей, получение семантического базиса в определении содержания информации, циркулирующей в системе; повторное использование знаний в ПрО; получение логической теории, на основе которой осуществляется вывод новых знаний, явно не заложенных в онтологии и расширяющих её функциональные возможности и класс решаемых задач.

ГМ используются в проектном управлении (например, [9]), в т.ч. в управлении человеческими ресурсами, управлении рисками проекта и т.д. Основная ценность применения онтологических моделей в проектном управлении заключается в том, что подобные модели могут стать основной частью базы знаний интеллектуальной системы поддержки принятия управленческих решений, а также основой системы управления знаниями проектной организации.

На основе результатов анализа работ, посвящённых онтологическому моделированию проектной работы [10-12], была разработана онтология проекта. Разработанный ГЗ реализован средствами редактора онтологий *Protégé*¹, фрагмент иерархии классов онтологии проекта представлен на рисунке 1.

Особенностями данного графа является включение в качестве класса ключевых слов для более подробного описания различных понятий ПрО (компетенций, результатов проекта, знаний, умений и владений работников и др.); указания конкретных составляющих понятий (конкретные методы, программы, методики и т.п.), что необходимо для уточнения критериев поиска исполнителя, более подробного описания средствами онтологий требований к работам проекта, автоматического определения связей между компетенциями, автоматического расширения набора ключевых слов компетенций.

Для определения групп оцениваемых компетенций использованы группы компетенций и компетенции инженеров, приведённые в работе [13].

При накоплении достаточного объёма данных о реализуемых проектах, частных работах, результатах научно-исследовательских проектов полученные результаты могут быть использованы при прогнозировании значимости компетенций при подготовке специалистов в соответствующей области, при корректировке соответствующих образовательных программ [14]. Особенно это актуально для ВУЗов, функционирующих в моногородах и обеспечивающих кадрами градообразующие предприятия.

В таблице 1 приведены примеры отношений между некоторыми классами онтологии проекта.

Таблица 1 – Примеры отношений между классами онтологии проекта

Класс-субъект	Отношение	Класс-объект
РаботаПроекта	ТребуетКомпетенцию	Компетенция
Проект	ВключаетРаботу	РаботаПроекта
Работник	ВладеетКомпетенцией	Компетенция
РаботаПроекта	ВключаетРазработку	МатематическаяМодель, КомпьютернаяМодель
ЦельПроекта, МатематическаяМодель	ОписываетОбъект	Объект
Проект	ИмеетЦель	ЦельПроекта, МатематическаяМодель
Работник	ВыполняетРаботу	РаботаПроекта
Проект	ИмеетОбъект	Объект
Компетенция	ЯвляетсяРодителемДля	Компетенция

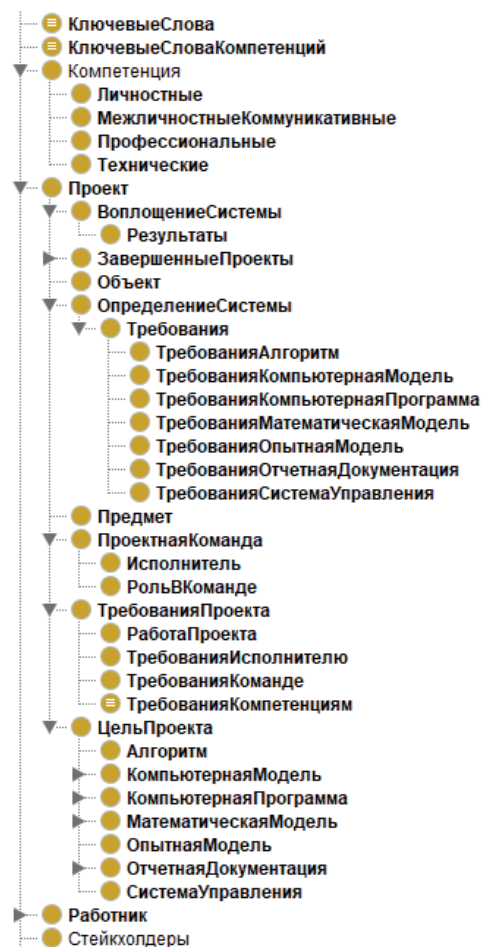


Рисунок 1 – Фрагмент иерархии классов онтологии проекта

¹ *Protégé* – A free, open-source ontology editor and framework for building intelligent systems. <https://protege.stanford.edu/>.

2 МО и анализ графов

Алгоритмы МО способны обучаться на наборах данных, описывающих исследуемые объекты или процессы и отражающие их свойства и закономерности [15, 16]. Для применения методов МО необходимо иметь векторные представления данных с действительными значениями, т.е. к ГМ можно применить существующие алгоритмы получения векторных представлений их вершин и отношений. Перечень некоторых задач, базирующихся на ГМ рассматриваемой ПрО, которые можно решить с применением методов МО [17], и некоторые возможные применения этих задач для проектного управления представлены ниже.

1) Предсказание связей:

- распределение работ (определение возможных исполнителей для работ нового проекта);
- определение новых связей между компетенциями и ключевыми словами;
- уточнение требований к профессиональной компетентности исполнителя нового проекта (определение новых связей между компетенциями и новой работой проекта).

2) Классификация вершин, сопоставление понятий:

- определение групп близких потенциальных исполнителей;
- формирование проектных групп, имеющих успешный опыт совместной работы;
- определение сходства (связей) между проектами и/или результатами проектов (разрабатываемыми моделями, алгоритмами, программами и др.);
- оценка соответствия компетенций потенциальных исполнителей проекта требованиям частных заданий.

Для построения векторных представлений вершин и отношений простого графа можно использовать такие алгоритмы, как *DeepWalk* [18], *Node2Vec* [19], нейросетевые алгоритмы и др. Существуют алгоритмы, которые позволяют каждой вершине и каждому отношению ГЗ ставить в соответствие уникальный вектор (*TransE*, *DistMult*, *RESCAL* и др.) [20]. Алгоритмы отличаются в т.ч. используемой функцией $f(x, r, y)$ оценки отношения r экземпляров x и y , которая позволяет оценить его правдоподобность; функцией потерь, значение которой минимизируется в процессе обучения модели. Некоторые алгоритмы имеют геометрическую интерпретацию (например, *TransE*), некоторые алгоритмы не позволяют работать с определёнными свойствами отношений (симметричность, антисимметричность и т.п.).

3 Анализ нового проекта с помощью ГМ и МО

3.1 Онтология и задача распределения работ

В качестве примера рассматривается задача определения возможных исполнителей работ по проекту, исходные данные, требования к исполнителям представлены в виде упрощённого оргграфа с типизированными рёбрами (рисунок 2). Данный граф является подграфом онтографа проекта (рисунок 1) и в качестве вершин включает экземпляры объектов ПрО (классов онтологии), а именно таких классов, как *Проект*, *Объект*, *МатематическаяМодель*, *Работник*, *РаботаПроекта*, *Компетенция*.

Граф включает завершённые проекты *Проект1* и *Проект2*. *Проект3* – это новый проект. Одной из целей *Проекта2* и *Проекта3* является построение некой модели одного и того же объекта *Объект1*.

На графе представлены требования к компетенциям исполнителя для каждой работы, преемственные связи между компетенциями. Имеется три потенциальных исполнителя для выполнения новой работы *Работа3*. Все возможные исполнители на высоком уровне владеют всеми компетенциями. *Работник3* не принимал участия ни в одном проекте.

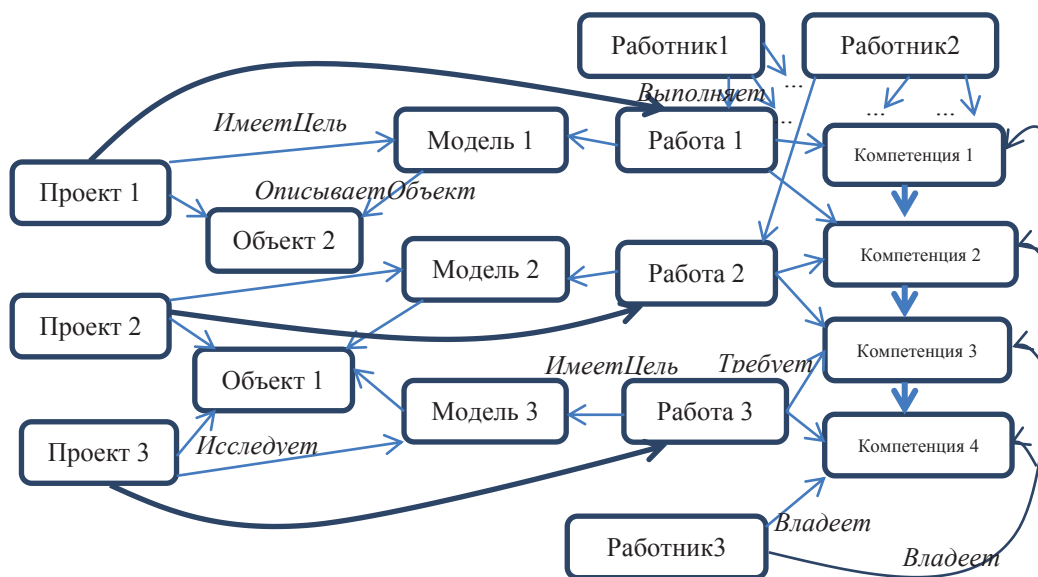


Рисунок 2 – Фрагмент упрощённого оргграфа для задачи распределения работы проекта

3.2 Меры семантической близости в онтологии и задача распределения работ

Для оценки сходства пар объектов в онтологической модели разработаны и используются различные меры семантической близости ($S(x, y) \in [0,1]$), которые позволяют использовать различные смысловые характеристики сравниваемых объектов (понятий, терминов), их свойства (атрибуты и отношения с другими объектами), анализ структуры и взаимное положение в онтологической иерархии [6].

Оценка соответствия работника требованиям к исполнителю работы проекта может быть определена через соотношение количества одинаковых свойств сравниваемых объектов ($A \cap B$, A и B – множества свойств этих объектов) к общему числу свойств этих объектов ($A \cup B$). В качестве сравниваемых объектов в рассматриваемой задаче могут быть выбраны уже выполненные работы (с известными исполнителями) и новая работа, цели этих работ (в примере это некие модели); ключевые слова (с учётом их преемственности), описывающие работы, модели, компетенции работников. Для поиска одинаковых и/или различных свойств можно построить *SPARQL*-запросы² и/или правила логического вывода *SWRL*³. Для автоматизации расчётов может быть использована библиотека *Python Owlready2*⁴ [21].

3.3 ГМ и задача распределения работ

Для применения алгоритмов получения векторных представлений для простых графов с однородными вершинами и однотипными отношениями необходимо сначала построить такой граф на основе онтологии. Общая схема построения графа и выбора проектной группы представлена на рисунке 3. Первым этапом является формирование неориентированного графа, вершинам которого соответствуют исполнители, а наличие связи между исполнителями означает их участие в одном и/или схожем уже реализованном проекте. Далее необходимо применить один из алгоритмов построения векторных представлений вершин простого графа. Полученные векторные представления потенциальных исполнителей затем могут быть использованы для формирования групп (кластеров) исполнителей с помощью извест-

² SPARQL 1.1 Query Language. <https://www.w3.org/TR/sparql11-query/>.

³ SWRL rules - Owlready2 documentation. <https://owlready2.readthedocs.io/en/latest/rule.html>.

⁴ Owlready2 documentation. <https://owlready2.readthedocs.io/en/latest/>.

ных алгоритмов кластеризации [16]. Далее осуществляется выбор кластера, исполнители в котором обладают всеми требуемыми компетенциями на достаточном уровне, для этого используются оценки компетенций исполнителей, а также их значимость (на рисунке 3: w_{ij} – оценка j -й компетенции i -го исполнителя, z_j – значимость j -й компетенции для проекта; $w_{ij}, z_j \in [0;1]$).

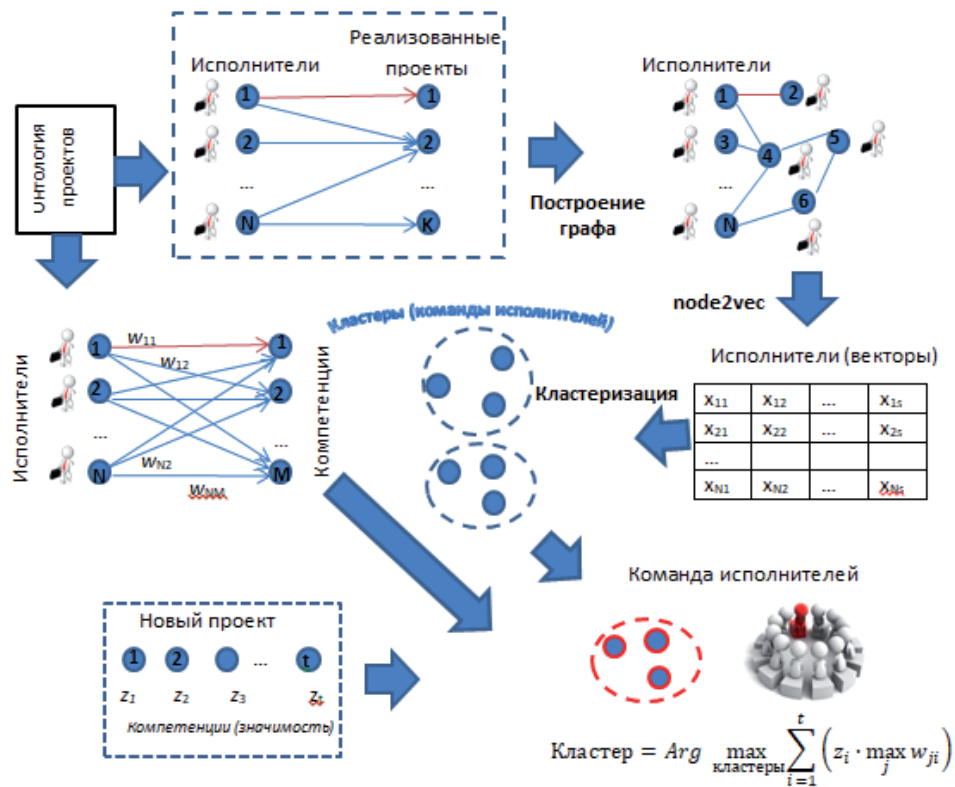


Рисунок 3 – Общая схема формирования графа исполнителей и выбора проектной группы

Описание содержания любого проекта на начальных этапах его жизненного цикла, как правило, всегда подразумевает описание множества работ, которые необходимо выполнить, взаимосвязей между работами и характеристик работ, в т.ч., набора компетенций исполнителей, необходимых для успешной реализации работ проекта. Набор компетенций работ проекта может быть определён на основе существующей модели компетенций сотрудников проектной организации, которая наряду с системами оценки и развития профессиональной компетентности персонала являются важной составляющей кадровой политики любой организации. Модель компетенций может быть построена по результатам экспертных опросов, анализа профессиональных стандартов, стандартов в области менеджмента проектной деятельности и т.п.

Ограничением рассмотренного подхода к формированию проектной группы является то, что новый исполнитель, возможно, самый компетентный, но не принявший участие ни в одном проекте, не будет иметь связей ни с кем.

Другой возможный подход состоит в том, что строится граф с вершинами, соответствующими потенциальным исполнителям и определяются связи между ними на основе анализа компетенций исполнителей и/или ключевых слов, которые описывают профессиональные знания, умения и навыки претендентов. В этом случае в кластеры попадут исполнители со

схожим уровнем профессиональной компетентности. Для получения векторных представлений вершин графа могут использоваться существующие функции языка *Python*⁵.

3.4 ГЗ и задача распределения работ

Пусть для заданного множества объектов ПрО E и множества отношений между этими объектами R построен ГЗ $K \subseteq E \times R \times E$, который содержит тройки $(h, r, t) \in K$; $h, t \in E$ представляют соответственно субъект («голова») и объект («хвост») отношения $r \in R$, которое описывает некоторое взаимодействие между объектами ПрО [20]. Задача предсказания связи определяется как предсказание субъекта или объекта для троек $(?, r, t)$ или $(h, r, ?)$ соответственно. Так, в примере на рисунке 2 определяется наиболее вероятный субъект в тройке $(?, \text{«ВыполняетРаботу»}, \text{«Работа3»})$.

Каждый алгоритм построения векторных представлений элементов графа использует некоторую функцию оценки отношения $f: E \times R \times E \rightarrow \mathbf{R}$, позволяющую оценить правдоподобность тройки $(h, r, t) \in K$. Эта оценка должна в идеале принимать максимальные значения для $(h, r, t) \in K$ и минимальные для $(h, r, t) \notin K$. Например, алгоритмы *RESCAL* и *DistMult*, основанные на факторизации разреженного тензора смежности ГЗ [17], моделируют объекты в виде векторов $h, t \in \mathbf{R}^d$, а отношения – в виде матриц $\mathbf{W}_r \in \mathbf{R}^{d \times d}$, в *DistMult* эти матрицы являются диагональными. Функция оценки отношения $(h, r, t) \in K$ задаётся формулой [20]:

$$f(h, r, t) = \mathbf{h}^T \mathbf{W}_r \mathbf{t} = \sum_{i=1}^d \sum_{j=1}^d w_{ij}^{(r)} h_i t_j.$$

Обучение модели, как правило, осуществляется путём минимизации некоторой функции потерь, так, в алгоритме *DistMult* функция потерь может иметь вид [22]:

$$L = \sum_{(h,r,t) \in K} \sum_{(h',r,t') \in K'} \max\{f(h',r,t') - f(h,r,t) + 1, 0\},$$

где K' – множество «негативных» троек:

$$K' = \{(h', r, t) \mid h' \in E, (h', r, t) \notin K\} \cup \{(h, r, t') \mid t' \in E, (h, r, t') \notin K\}.$$

В качестве оптимизатора модели может быть использован мини-пакетный стохастический градиентный спуск [22].

Для работы с ГЗ использовались готовые функции библиотеки *Python PyKEEN*⁶. После обучения модели (например, *DistMult*, *TransE*, *ConvE*, *RotatE*) получены оценки различных отношений $f(x, r, y)$ экземпляров x и y , чем выше оценка, тем больше вероятность того, что рассматриваемое отношение окажется верным. При этом минимальное значение оценки, чтобы отношение было определено как точно верное, отсутствует.

Были смоделированы различные требования к работам нового проекта и с помощью векторных представлений вершин графа получены оценки вероятности связи между вершиной, соответствующей новой работе, и вершинами, соответствующими потенциальным исполнителям. Для ситуации, представленной на рисунке 2, по результатам работы большинства алгоритмов лучшим исполнителем является работник *Работник2*. Это объясняется тем, что хотя все потенциальные исполнители владеют требуемыми компетенциями, но *Работник2* имеет опыт выполнения подобных работ, а именно опыт разработки некой модели объекта *Объект1*.

⁵ Python3 implementation of the node2vec algorithm. <https://github.com/eliorc/node2vec>.

⁶ PyKEEN 1.10.0 documentation. <https://pykeen.readthedocs.io/en/stable/index.html>.

Заключение

В работе приведены возможные подходы к применению ГМ при решении задачи распределения работ проекта между потенциальными исполнителями. Рассмотрены онтологии, ГЗ и простые графы. Выделены возможные направления применения векторного представления ГЗ и простых графов в проектном управлении. Средствами редактора онтологий *Protégé* реализована онтология проекта. Представление элементов графа в виде числовых векторов позволяет применить к исследованию ПрО методы МО. На примере решения задачи формирования рабочей группы проекта осуществлён выбор исполнителя проектной работы. Все расчёты осуществлены средствами соответствующих функций библиотек *Owlbready2* и *PyKEEN* языка *Python*. Полученные результаты не противоречат ожидаемым экспертным оценкам.

Список источников

- [1] **Царьков И.Н.** Математические модели управления проектами. Москва: ИНФРА-М, 2019. 514 с.
- [2] **Асанов А.З., Мышкина И.Ю.** Процедура формирования команды исполнителей проекта на основе когнитивных карт и генетических алгоритмов // Проблемы управления и моделирования в сложных системах: труды XXI Международной конференции. Самара : Офорт, 2019. С. 354–358.
- [3] **Сукар Л.Э.** Вероятностные графовые модели. Принципы и приложения. Москва: ДМК Пресс, 2021. 338 с.
- [4] **Уилсон Робин.** Введение в теорию графов, 5-е изд. Санкт-Петербург: ООО "Диалектика", 2019. 240 с.
- [5] **Кудрявцев Д.В., Беглер А.М., Гаврилова Т.А., Лещева И.А., Кубельский М.В., Тушканова О.Н.** Метод коллективной визуальной разработки онтологического графа знаний // Искусственный интеллект и принятие решений. 2019. №1. С. 27-38.
- [6] **Крюков К.В., Панкова Л.А., Пронина В.А., Суховеров В.С., Шипилина Л.Б.** Меры семантической близости в онтологии // Проблемы управления. 2010. №5. С.2-14.
- [7] **Цуканова Н.И.** Онтологическая модель представления и организации знаний. Москва: Горячая линия – Телеком, 2015. 272 с.
- [8] **Гаврилова Т.А., Хорошевский В.Ф.** Базы знаний интеллектуальных систем. Санкт-Петербург: Питер, 2020. 384 с.
- [9] **Черняховская Л.Р., Малахова А.И.** Разработка моделей и методов интеллектуальной поддержки принятия решений на основе онтологии организационного управления программными проектами // Онтология проектирования. 2013. №4. С. 42-52.
- [10] **Левенчук А.** Системноинженерное мышление. TechInvestLab, 2 апреля 2015. 305 с. [systems_engineering_thinking_2015.pdf](https://techinvestlab.ru/systems_engineering_thinking_2015.pdf) (techinvestlab.ru).
- [11] **Zaoua Wiem, Arfa Rabai Latifa Ben, Alalyani Wafa Rashid.** Towards an Ontology Based-Approach for Human Resource Management // The 10th International Conference on Ambient Systems, Networks and Technologies (ANT). April 29 – May 2, 2019. Leuven, Belgium. P.417-424.
- [12] **Podgorelec V., Grasic B.** Implementing Innovative IT Solutions with Semantic Web Technologies // Products and Services; from R&D to Final Solutions. Nov., 2010. DOI: 10.5772/10399.
- [13] **Макарова И.В., Хабибуллин Р.Г., Ушенин А.М., Михеева С.А., Карабцев В.С.** Компьютер в инженерном образовании: новые возможности в подготовке инженеров для креативной экономики // Инженерное образование. 2016. №20. С.72-79.
- [14] **Асанов А.З., Мышкина И.Ю., Грудцына Л.Ю.** Прогнозирование востребованности компетенций при корректировке программ обучения с помощью когнитивных моделей // Онтология проектирования. 2019. Т.9. №2 (32). С.203–213. DOI: 10.18287/2223-9537-2019-9-2-203-213.
- [15] **Болотова Л.С.** Системы искусственного интеллекта: модели и технологии, основанные на знаниях. Москва: Финансы и статистика, 2012. 664 с.
- [16] **Флах П.** Машинное обучение. Москва: ДМК, 2015. 400 с.
- [17] **KG Course 2021.** Курс по графам знаний (Knowledge Graphs) и как их готовить в 2021 году. На русском языке. <https://migalkin.github.io/kgcourse2021/>.
- [18] **Perozzi B., Al-Rfou R., Skiena S.** DeepWalk: Online learning of social representations // KDD '14: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. August 2014. Pages 701–710. DOI: 10.1145/2623330.2623732.
- [19] **Grover Aditya, Leskovec Jure.** node2vec: Scalable Feature Learning for Networks // KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. August 2016. P.855–864. <https://doi.org/10.1145/2939672.2939754>.

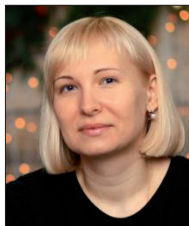
- [20] *Ali M., Berrendorf M., Hoyt C. T., Vermue L., Galkin M., Sharifzadeh S., Fischer A., Tresp V., Lehmann J.* Bringing Light Into the Dark: A Large-scale Evaluation of Knowledge Graph Embedding Models Under a Unified Framework. 2020. DOI: 10.48550/arXiv.2006.13365.
- [21] *Jean-Baptiste Lamy.* Ontologies with Python Programming OWL 2.0 Ontologies with Python and Owlready2. Apress, 2021. 270 p.
- [22] *Yang Bishan, Yih Wen-tau, He Xiaodong, Gao Jianfeng, Deng Li.* Embedding Entities and Relations for Learning and Inference in Knowledge Bases // ICLR'15: Proceedings of the 3rd International Conference on Learning Representations. USA. May 7-9, 2015. <https://arxiv.org/abs/1412.6575>.

Сведения об авторах



Асанов Асхат Замилович, 1949 г. рождения. Окончил Казанский государственный университет в 1972 г., д.т.н. (2004), профессор. Профессор кафедры автоматических систем Института искусственного интеллекта МИРЭА–Российского технологического университета. В списке научных трудов более 250 работ в области системного анализа, управления и обработки информации. ORCID: 0000-0001-6724-0816; Author ID (РИНЦ): 263248; Author ID (Scopus): 7003438319. asanov@mirea.ru ✉.

Мышкина Ирина Юрьевна, 1980 г. рождения. Окончила Елабужский государственный педагогический институт в 2002 г., к.т.н.(2018). Доцент кафедры системного анализа и информатики Набережночелнинского института (филиала) Казанского (Приволжского) федерального университета. В списке научных трудов более 30 работ в области управления в социальных и экономических системах. ORCID: 0000-0003-4309-3350; Author ID (РИНЦ): 603791; Author ID (Scopus): 57189268396; Researcher ID (WoS): M-4899-2015. mirinau@mail.ru.



Грудцына Лариса Юрьевна, 1980 г. рождения. Окончила Елабужский государственный педагогический институт в 2002 г. Старший преподаватель кафедры системного анализа и информатики Набережночелнинского института (филиала) Казанского (Приволжского) федерального университета. В списке научных трудов более 20 работ в области управления в социальных и экономических системах. ORCID: 0000-0003-1307-6605; Author ID (РИНЦ): 804837; Author ID (Scopus): 57189270650; Researcher ID (WoS): M-6037-2015. larisa_u_g@mail.ru.

Поступила в редакцию 01.03.2023, после рецензирования 26.06.2023. Принята к публикации 28.06.2023.



Scientific article

DOI: 10.18287/2223-9537-2023-13-2-232-242

Application of graph models in project management

© 2023, A.Z. Asanov¹✉, I.Yu. Myshkina², L.Yu. Grudtsyna²

¹ Institute of Cybernetics of MIREA, Russian Technological University, Moscow, Russia

² Naberezhnye Chelny Institute (branch) of Kazan (Volga Region) Federal University, Naberezhnye Chelny, Russia

Abstract

The possibility of using graph models in project management to solve the problem of forming a project team and distributing project work among potential performers is investigated. The possible areas of application of ontological modeling in project management are highlighted. The ontology of the project is implemented by means of the Protégé ontology editor. The possibility of constructing vector representations of elements of simple graphs and knowledge graphs in the problem of forming a project team is investigated. The main objectives of the research are: the construction of graph models of projects, a simple graph and a knowledge graph; the use of vector representations of vertices and rela-

tionships of graph models to solve the problem of distributing project work. The possibility of applying semantic proximity measures in ontology is analyzed. The novelty of the proposed approach lies in the use of data on participants in completed projects when forming a project team, which increases the chance of successful project implementation. The representation of graph elements in the form of numerical vectors makes it possible to apply machine learning methods to the study of the subject area. An example of solving the problem of forming a project working group is given, in which the executors of the project work are selected. All calculations were carried out using the Python language.

Keywords: *graph model, knowledge graph, ontology, project, team, performer.*

For citation: *Asanov AZ, Myshkina IY, Grudtsyna LYu.* Application of graph models in project management [In Russian]. *Ontology of designing.* 2023; 13(2): 232-242. DOI: 10.18287/2223-9537-2023-13-2-232-242.

Conflict of interest: The author declares no conflict of interest.

List of figures and table

Figure 1 - A fragment of the project ontology class hierarchy

Figure 2 - A fragment of a simplified digraph for the task of distributing project work

Figure 3 - The general scheme for forming the graph of performers and the selection of the project team

Table 1 - Examples of relations between project ontology classes

References

- [1] *Tsarkov IN.* Mathematical models of project management [In Russian]. Moscow: INFRA-M, 2019. 514 p.
- [2] *Asanov AZ, Myshkina IYu.* The procedure for forming a team of project performers based on cognitive maps and genetic algorithms [In Russian]. Problems of control and modeling in complex systems: Proceedings of the XXI International Conference. Samara: Ofort, 2019. P.354–358.
- [3] *Sucar LE.* Probabilistic graphical models. Principles and applications [In Russian]. Moscow: DMK Press, 2021. 338 p.
- [4] *Robin JW.* Introduction to GRAPH THEORY, fifth edition. Saint-Petersburg: LLC "Dialektika", 2019. 240 p.
- [5] *Kudryavtsev DV, Begler AM, Gavrilova TA, Leshcheva IA, Kubelskiy MV, Tushkanova ON.* Method for collaborative visual creation of a knowledge graph [In Russian]. *Artificial Intelligence and Decision Making.* 2019; 1: 27-38.
- [6] *Kryukov KV, Pankova LA, Pronina VA, Suhoverov VS, Shipilina LB.* Semantic similarity measures in ontology. Review and classification [In Russian]. *CONTROL SCIENCES.* 2010; 5: 2-14.
- [7] *Tsukanova NI.* Ontological model of knowledge representation and organization [In Russian]. Moscow: Goryachaya Liniya – Telekom, 2015. 272 p.
- [8] *Gavrilova TA, Khoroshevsky VF.* Knowledge bases of intelligent systems [In Russian]. Saint-Petersburg: Piter, 2020. 384 p.
- [9] *Chernyakhovskaya LR, Malahova AI.* Development of intellectual decision support models and methods based on ontology of software projects organization management [In Russian]. *Ontology of designing.* 2013; 4: 42-52.
- [10] *Levenchuk A.* Systems engineering thinking. TechInvestLab, 02.04.2015. 305 p. [systems_engineering_thinking_2015.pdf](https://techinvestlab.ru/systems_engineering_thinking_2015.pdf) (techinvestlab.ru).
- [11] *Zaouga Wiem, Arfa Rabai Latifa Ben, Alalyani Wafa Rashid.* Towards an Ontology Based-Approach for Human Resource Management // The 10th International Conference on Ambient Systems, Networks and Technologies (ANT). April 29 – May 2, 2019. Leuven, Belgium. P. 417-424.
- [12] *Podgorelec V, Grasic B.* Implementing Innovative IT Solutions with Semantic Web Technologies // Products and Services; from R&D to Final Solutions. Nov., 2010. DOI: 10.5772/10399.
- [13] *Makarova IV, Khabibullin RG, Ushenin AM, Mikheeva SA, Karabtsev VS.* Computer in engineering education: new opportunities in training engineers for the creative economy [In Russian]. *Engineering education.* 2016; 20: 72-79.
- [14] *Asanov AZ, Myshkina IYu, Grudtsyna LYu.* Forecasting the demand for competencies when adjusting training programs using cognitive models [In Russian]. *Ontology of designing.* 2019; 9(2): 203–213. DOI: 10.18287/2223-9537-2019-9-2-203-213.
- [15] *Bolotova LS.* Artificial intelligence systems: knowledge-based models and technologies [In Russian]. Moscow: Finansy I Statistika, 2012. 664 p.
- [16] *Flach P.* Machine learning. Moscow: DMK, 2015. 400 p.
- [17] KG Course 2021. <https://migalkin.github.io/kgcourse2021/>.

- [18] **Perozzi B, Al-Rfou R, Skiena S.** DeepWalk: Online learning of social representations // KDD '14: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. August 2014. Pages 701–710. DOI: 10.1145/2623330.2623732.
 - [19] **Grover Aditya, Leskovec Jure** node2vec: Scalable Feature Learning for Networks // KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. August 2016. Pages 855–864. DOI: 10.1145/2939672.2939754.
 - [20] **Ali M, Berrendorf M, Hoyt CT, Vermue L, Galkin M, Sharifzadeh S, Fischer A, Tresp V, Lehmann J.** Bringing light into the dark: A large-scale evaluation of knowledge graph embedding models under a unified framework. 2020. DOI: 10.48550/arXiv.2006.13365.
 - [21] **Jean-Baptiste Lamy.** Ontologies with Python Programming OWL 2.0 Ontologies with Python and Owlready2. Apress, 2021. 270 p.
 - [22] **Yang Bishan, Yih Wen-tau, He Xiaodong, Gao Jianfeng, Deng Li.** Embedding Entities and Relations for Learning and Inference in Knowledge Bases // ICLR'15: Proceedings of the 3rd International Conference on Learning Representations. USA. May 7-9, 2015. <https://arxiv.org/abs/1412.6575>.
-

About the authors

Askhat Zamilovich Asanov (b. 1949) graduated from Kazan State University in 1972, D. Sc. Eng. (2004), Professor. He is a Professor of the Automatic Systems Department in the Institute of Artificial Intelligence of MIREA, Russian Technological University. He is the author and a co-author of more than 250 publications in the field of system analysis, control and information processing. ORCID: 0000-0001-6724-0816; Author ID (РИИЦ): 263248; Author ID (Scopus): 7003438319. asanov@mirea.ru. ✉.

Irina Yurievna Myshkina (b. 1980) graduated from the Elabuga State Pedagogical Institute in 2002, Ph.D. (2018). Associate Professor of the Department of Systems Analysis and Informatics of the Naberezhnye Chelny Institute (branch) of Kazan (Volga Region) Federal University. In the list of scientific papers there are more than 30 works in the field of management in social and economic systems. ORCID: 0000-0003-4309-3350; Author ID (РИИЦ): 603791; Author ID (Scopus): 57189268396; Researcher ID (WoS): M-4899-2015. mirinau@mail.ru.

Larisa Yurievna Grudtsyna (b. 1980) graduated from the Elabuga State Pedagogical Institute in 2002. Senior Lecturer at the Department of System Analysis and Informatics of the Naberezhnye Chelny Institute (branch) of Kazan (Volga Region) Federal University. In the list of scientific papers there are more than 20 works in the field of management in social and economic systems. ORCID: 0000-0003-1307-6605; Author ID (РИИЦ): 804837; Author ID (Scopus): 57189270650; Researcher ID (WoS): M-6037-2015. larisa_u_g@mail.ru.

Received March 01, 2023. Revised June 26, 2023. Accepted June 28, 2023.
