

FOAF に基づく SNS の統一ソーシャルグラフモデルの提案

M2009MM032 横井 公紀

指導教員 青山 幹雄

1. はじめに

SNS(Social Network Service)の利用が拡大している[1].しかし、各 SNS が基礎とするソーシャルグラフの構造は異なる。従って、SNS 間でソーシャルグラフの相互運用性が保証されず、複数の SNS 上の日記等の活動記録の統一的管理、異なる SNS 間のユーザの関係構築、ソーシャルアプリケーション開発が困難である。

本研究では SNS のソーシャルグラフの構造に着目し、構造の差異を吸収してマッピングする統一ソーシャルグラフモデル GSGM(Generic Social Graph Model)を提案する。Facebook, Twitter などの SNS に適用し、有効性を評価する。

2. SNS のソーシャルグラフの相互運用性問題

ソーシャルグラフの相互運用性問題(図 1)について、構造とインタフェースの視点から述べる。

(1) ソーシャルグラフの構造の差異

SNS 間でソーシャルグラフを構成するノードとエッジの属性の種類と、そのセマンティクスが異なることである。SNS で固有の属性が定義されたソーシャルグラフは、相互運用性が保証されない。また、属性名が同じでも、セマンティクスが異なれば相互運用性が保証されない。

(2) ソーシャルグラフのインタフェースの差異

ソーシャルグラフを利用するための Web API の仕様が異なることである。仕様の差異により、SNS の Web API の仕様毎にソーシャルアプリケーション開発が必要となる。従って、各仕様の学習コスト、開発コストが増加する。

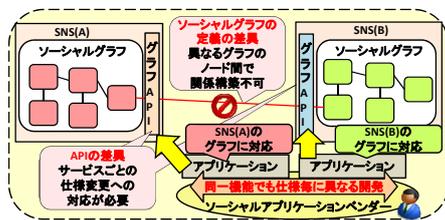


図1 ソーシャルグラフの相互運用性問題

3. 関連研究

3.1. ソーシャルグラフの相互運用性に関する研究

特定の SNS に依存しない形式で全ての SNS のソーシャルグラフを統合するグローバルソーシャルグラフと、実現に必要な API 群が提案されている[3].しかし、ソーシャルグラ

フの構造は議論されていない。

3.2. OpenSocial

ソーシャルグラフの共通 API 群と共通プラットフォームである OpenSocial を採用する SNS が増加している。各 SNS 間でソーシャルグラフとソーシャルアプリケーションの相互運用性が保証されるが、MySpace のように独自定義を採用した SNS では保証されない。

3.3. FOAF (Friend Of A Friend)

FOAF は、Web 上の人物の特徴と人間間の関係の連鎖を表現するモデルである[2]. FOAF は特定のプラットフォームに非依存である。また、RDF を利用して FOAF を記述するためのスキーマが提案されている。

4. アプローチ

ソーシャルグラフの相互運用性問題を、FOAF の拡張と複数 SNS のソーシャルグラフの統合による統一ソーシャルグラフのモデル化により解決する(図 2).

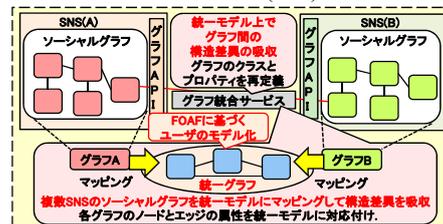


図2 本研究のアプローチ

(1) FOAF の拡張

SNS ユーザの特徴とユーザ間の関係を十分に定義する目的で、FOAF を拡張する。FOAF に基づきユーザとその間の関係をモデル化することで、特定の SNS に非依存なソーシャルグラフを実現できる。

(2) 複数 SNS のソーシャルグラフの統合

構造とインタフェースが異なる複数の SNS のソーシャルグラフを統一的に表現するソーシャルグラフモデルを定義する。統一ソーシャルグラフのインタフェースから、特定のソーシャルグラフへの依存部分を分離して構造を定義する。

5. FOAF に基づく GSGM のモデル化

5.1. GSGM の目的

複数のソーシャルグラフの構造とインタフェースの差異を吸収した統一ソーシャルグラフの実現が目的である。GSGM は、属性のインタフェースを定義する抽象定義 (AbstractGSGM) と、実際の SNS のソーシャルグラフの属性

定義と AbstractGSGM を対応付ける具体定義 (ConcreteGSGM)から構成される(図3).

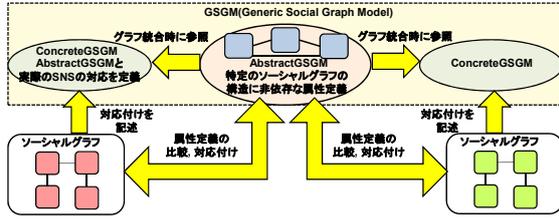


図3 GSGMの構成

5.2. モデル化の前提条件

(1) 属性抽出の順序

- 1) Facebook, OpenSocial の各ソーシャルグラフを取得する API の仕様書を分析し、各ソーシャルグラフの属性の共通部分を抽出する。
- 2) FOAF と(1)の共通部分と、個別部分に分類する。

(2) ソーシャルグラフ構造の前提条件

GSGMは一方方向の有向グラフとし、ノードが SNS ユーザ, エッジがノード間の関係を表す。

5.3. FOAF に基づく GSGM メタモデル

(1) GSGM メタモデルの構成 (図4)

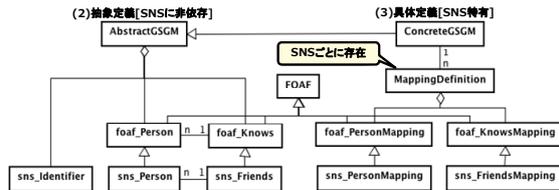


図4 GSGM(Generic Social Graph Model)

(2) 抽象定義 AbstractGSGM

ソーシャルグラフのノードとエッジの属性名と型のみを定義する(図5)。以下、各要素の詳細を示す。

- A) foaf_Person: FOAF に基づきノードの属性を定義。
- B) sns_Person: FOAF を拡張してノードの属性を定義。
- C) foaf_Knows: FOAF に基づきエッジの属性を定義。
- D) sns_Friends: foaf_Knows の定義に加え、関係の起源を表す SNS 名, 知人名, 知人の識別子を定義する。個人が複数の SNS で持つ関係を同一グラフ上に表現するために必要である。
- E) sns_Identifier: ノードの識別子, 利用するソーシャルアプリケーション ID を表す属性を定義。

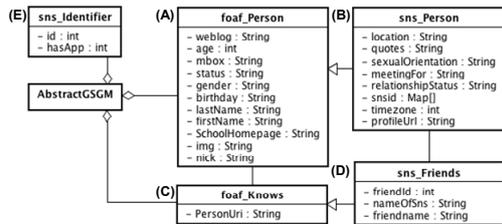


図5 AbstractGSGM

(3) 具体定義 ConcreteGSGM

実在する SNS のソーシャルグラフの属性定義と AbstractGSGM との対応付けを定義する(図6)。各クラスの属性値として、対応するソーシャルグラフの属性名を与える。以下、各要素の詳細を示す。

- a) foaf_PersonMapping: foaf_Person のセマンティクスに対応する各ソーシャルグラフの具体的な属性定義。
- b) sns_PersonMapping: sns_Person のセマンティクスに対応する各ソーシャルグラフの具体的な属性定義。
- c) foaf_KnowsMapping: foaf_Knows との対応付けを定義する。PersonUri に、統一ソーシャルグラフの知人ノードの URI を与える。
- d) sns_FriendsMapping: sns_Friends との対応付けを定義する。friendId に統一ソーシャルグラフの知人ノードの ID を示す属性名(sns_Identifier の id), nameOfSns に関係の起源の SNS 名, friendname に知人ノードの表示名を表す属性名(foaf_Person の name)を与える。

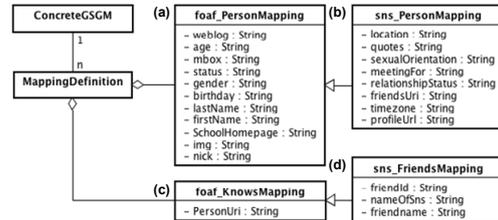


図6 ConcreteGSGM

5.4. 複数 SNS から GSGM へのマッピング方法

複数 SNS のソーシャルグラフの属性定義を GSGM に変換し、マッピングする方法を提案する。

(1) 複数 SNS と AbstractGSGM の対応付け

AbstractGSGM と各ソーシャルグラフの属性との対応を ConcreteGSGM に定義する。ConcreteGSGM には、AbstractGSGM の定義に基づき、各ソーシャルグラフと FOAF との共通部分と個別部分を MappingDefinition に記述する。さらに AbstractGSGM から各 ConcreteGSGM への参照を、ConcreteGSGMReference で記述する(図7)。

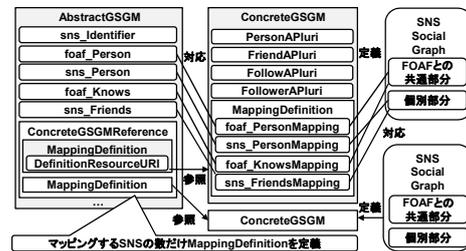


図7 GSGM を定義する構造

(2) AbstractGSGM の定義

ソーシャルグラフの属性名と型を表すインタフェースのみを定義し、特定の SNS に依存する情報を持たない。SNS 固有の情報を持つ ConcreteGSGM を ConcreteGSGM Reference で参照し、マッピング対象のソーシャルグラフとその組合せを MappingDefinition で定義する。

MappingDefinition は、SNS 毎の ConcreteGSGM のリソースの URI を DefinitionResourceURI として参照を宣言する。対象の SNS の数だけ、MappingDefinition を定義する(図 8)。



図 8 AbstractGSGM を定義する構造

(3) ConcreteGSGM の定義

AbstractGSGM の属性定義と実際の SNS のソーシャルグラフの属性定義の対応付けと、各 SNS のソーシャルグラフを利用する Restful API の URI を定義する(図 9)。



図 9 ConcreteGSGM を定義する構造

ConcreteGSGM のノードの各属性値を決定する方法を図 10 に示す。AbstractGSGM の各属性に対応するセマンティクスを持つ各 SNS のソーシャルグラフの属性名を、対応する ConcreteGSGM の各属性値として与える。

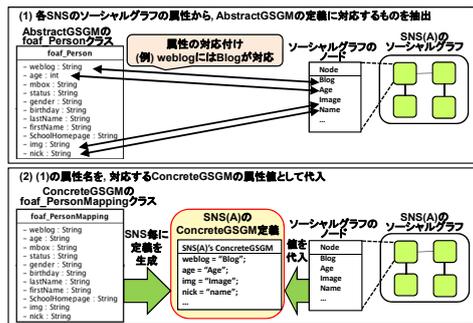


図 10 ConcreteGSGM のノードの属性値の定義方法

また、ConcreteGSGM のエッジの各属性値を次のように決定する。foaf_KnowsMapping の PersonUri の属性値に、生成する統一ソーシャルグラフのノードの URI を表す属性名を与える。さらに、foaf_PersonMapping の nameOfSns に関係の起源を表す SNS 名を表す属性名、friendId に AbstractGSGM で定義した識別子、friendname に名前を表すノードの属性名を与える。

5.5. GSGM を操作する統一 Web API

Web API は、ソーシャルグラフの構造から、取得範囲と探索方法に基づき定義する必要がある。

(1) 取得範囲

ソーシャルグラフ上の人物あるいはその知人の単位で属性を取得できる必要がある。従って、取得範囲は人物とその直接の知人の範囲となる。人物を取得する API と、その知人を取得する API の二種類を定義する。

(2) 探索方法

ソーシャルグラフの属性に基づいた探索方法の決定が必要である。従って、AbstractGSGM の属性に対応する Web API のパラメータを定義し、各属性値をソーシャルグラフのフィルタリングに用いる。フィルタリングとは、パラメータの属性値をキーとしてソーシャルグラフを探索し、一致するノードを抽出した部分グラフを生成することである。ユーザ ID を必須パラメータとし、ユーザのノードとその直接の知人までの部分グラフを返す。ユーザ ID と AbstractGSGM の属性値をパラメータに渡すと、直接の知人までの範囲内で属性値によってフィルタリングした部分を返す。

統一 Web API として、取得範囲と探索方法に基づき getGSGMUserNode と getGSGMFriendNode の二種類を定義した(図 11)。

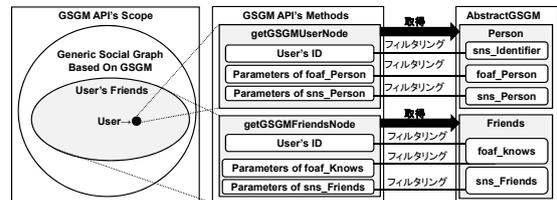


図 11 統一 Web API の構造

5.6. ソーシャルグラフ統合サービス

本研究では GSGM を XML で記述する。そのための XML 言語仕様 GSGL(Generic Social Graph Language)を提案する。また、5.4 の方法を用いて複数の SNS が持つ特定のユーザを中心ノードとするソーシャルグラフを統合するソーシャルグラフ統合サービスを提案する。

(1) GSGM を記述する XML 言語仕様 GSGL

GSGL では、GSGM の構造を XML の木構造で定義する。属性の型を XML スキーマにビルトインされているデータ型を利用して定義する。さらに、ConcreteGSGL の参照を RDF スキーマの rdf:resource を利用して定義する。AbstractGSGM を XML 形式で記述するための言語仕様を AbstractGSGL、ConcreteGSGM の言語仕様を ConcreteGSGL と呼ぶ。

(2) ソーシャルグラフ統合サービス

各 SNS のソーシャルグラフを統合し、統一ソーシャルグラフを生成する(図 12)。ConcreteGSGL を用いて各 SNS のソーシャルグラフを取得し、ユーザとその知人のノードを生成

する。各ノードをエッジの属性定義で関連付けて、統一ソーシャルグラフを生成する。

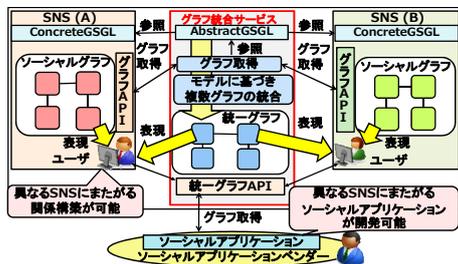


図 12 ソーシャルグラフ統合サービス

6. 実際の SNS への適用

GSGL を、ソーシャルグラフの構造と API の定義が異なる Facebook, MySpace, Twitter に適用する。

(1) ConcreteGSGL の定義

各 SNS の API の仕様書から、各ソーシャルグラフと AbstractGSGL の対応を ConcreteGSGL.xml に定義する。

(2) AbstractGSGL の定義

各 SNS の ConcreteGSGL.xml の URI を ConcreteGSGL Reference に定義した AbstractGSGL.xml を生成する。

7. 評価と考察

7.1. 実際の SNS への適用による評価と考察

(1) GSGL に変換可能な属性数 (表 1)

表 1 GSGL に変換可能な各ソーシャルグラフの属性数

ソーシャルグラフ名	ノード				エッジ			
	人物を表す属性数	FOAFとの共通属性数	個別属性数	非対応属性数	関係を表す属性数	FOAFとの共通属性数	個別属性数	非対応属性数
GSGL	19	11	8	-	4	1	3	-
Facebook	24	11	8	5	4	1	3	0
MySpace	35	10	8	17	4	1	3	0
Twitter	10	3	4	3	4	1	3	0

ノードの属性は、Facebook の 79%, MySpace の 51%, Twitter の 70%が変換可能であった。また、エッジの属性により、各 SNS の関係を満足にマッピング可能であった。

変換不可能な属性は、各 SNS に依存しているか、その SNS しか定義していない属性であり、これらの属性を含むソーシャルグラフは SNS 間の相互運用性が保証できない。また、変換不可能な属性が含まれるのはノードのみであり、それらの属性を分析した結果、sns_Person の name 属性のように、ユーザ間の関係の表現に必要な属性は含まれなかった。従って、変換不可能な属性は、ユーザ間の関係の表現に必要ではないことが分かった。

以上から、異なる構造の複数 SNS のユーザ間の関係の表現に必要な属性を GSGL で定義して関係の構築が可能と言え、モデルの有効性を確認できた。

(2) 統一 Web API で取得可能なユーザの知人関係

統一 Web API を用いて、異なるインタフェースの SNS のソーシャルグラフの知人関係を統一的な仕様に基き取

得可能となった。SNS 毎の知人関係を sns_Friends の nameOfSns の属性値でフィルタリングして取得可能である。

また、統一 Web API のパラメータは AbstractGSGL の各クラスの属性に対応する。従って、GSGL に変換可能な各 SNS のソーシャルグラフの属性(表 1)を、人物とその知人関係の単位で取得可能である。

(3) 実在する SNS のソーシャルグラフの相対的評価

GSGL を基準とし、複数の SNS のソーシャルグラフの比較と評価が可能である。(1)より、SNS 毎に表現可能な属性数が異なるため、GSGL を用いて実在するソーシャルグラフ間の相対的な属性数の比較が可能になる。例えば表 1 より、Facebook と MySpace が多くの個別属性を定義していること、Twitter には人物を表す情報が不足していることが分かる。

7.2. 関連研究との比較による評価と考察

文献[3]では、相互運用可能なソーシャルグラフの具体的な構造は未定義であった。提案した GSGL では、複数の SNS のソーシャルグラフの異なるノードとエッジの構造を、AbstractGSGL と ConcreteGSGL の二階層に分離して吸収した。この結果、異なる SNS のユーザ間の関係を各ソーシャルグラフの構造によらず統一的に表現可能である。

また、関連研究で提案された API 群と、本研究で提案した GSGL を組み合わせ、複数 SNS 間のユニークな人間関係を統一的な形式で表現可能である。

8. 今後の課題

(1) 複数ソーシャルグラフのノードの等価性の判断基準と方法

Web 上のユニークな人間関係を表すため、複数 SNS のソーシャルグラフのノードの等価性の判断基準と、判断方法について議論が必要である。

(2) ソーシャルグラフ統合サービスのアーキテクチャ

各 SNS のソーシャルグラフを利用するための API には、認証やプロトコルの制約がある。これらの条件を考慮した具体的なアーキテクチャが必要である。

9. まとめ

FOAF を拡張し、異なる構造のソーシャルグラフを基礎とする複数の SNS にまたがるユーザ間の関係構築を可能にする統一ソーシャルグラフモデル GSGL を提案した。提案方法を実際の SNS に適用し、有効性を示した。

参考文献

- [1] 青山 幹雄, スマートソーシャルサービスシステムの構想, 情報処理学会ウィンターワークショップ 2011・イン・修善寺 論文集, Jan. 2011, pp. 55-56.
- [2] D. Brickley, et al., FOAF Vocabulary Specification, 2010, <http://xmlns.com/foaf/spec/>.
- [3] B. Fitzpatrick, Thoughts on the Social Graph, 2010, <http://www.bradfitz.com/social-graph-problem/>.