

# Requirements Extraction from User Feedback On The Basis Of Ontology

Wajid Ahmed Channa<sup>1</sup>, Qamar Uddin Khand<sup>2</sup>, Sajid Ahmed Ghanghro<sup>3</sup>

<sup>1,2,3</sup>Department of Computer Science, Sukkur IBA University Sukkur, Sindh, Pakistan

## Abstract

The study is designed to scrutinize mobile app ratings for requirements extraction as well as to check quality of an app through the exploitation of interleaved and independent mobile app rating schema, which comprises of ratings and reviews. However, a developer intends to collect app ratings for gathering its requirements because of throwing a newer version towards its users over an app store, which provides them easiness in all aspects. An ontological strategy is the best way to evaluate the accurate analysis of app ratings by interpreting them in such a proposed interface, which actually enumerates best results by putting a good strength of its schema. Hence, a developer can easily extract requirements with a complete rigorousness, so also it becomes less challenging to satisfy the users. The study goes through the ontological phenomenon throughout the complete scenario of requirements extraction, which a developer can easily put them in practice.

## Keywords

*App rating; Google Play store; NLP; Feedback; Requirements extraction;*

## 1. Introduction

In recent modern age, the field of mobile applications has emerged much enough because of huge and growing number of mobile app downloads and their usage on different sort of mobiles available in the market, besides it, there is a huge ratio of failure of such apps on mobiles, which is increasing enormously day-by-day due to several reasons; which are highlighted later on but only major reasons of them, are followed and discussed by this proposed research study to avoid such failures, no doubt many studies are conducted with regard to failure of these apps and to know reasons about failure of apps but some of these studies so far succeeded in the domain of mobile apps.

However, these app stores provide two types of apps; paid and unpaid or free apps to their users; as some of these mobile apps are thrown as open source, so that a user can easily download such apps without paying any kind of cost, besides them, others are paid, which require online payment before going to get them downloaded. It is found in the literature, so also in real environment that the recent market of mobile phones, is using a few dedicated apps distributing platforms on concrete app stores, through which such apps are provided to a mobile user for

facilitating them to fulfill different tasks and to achieve their goals, the popular play stores are mentioned as under;

1. Google Play Store
2. Apple App Store
3. Blackberry World Store
4. Microsoft Phone Apps Store

Besides them, some of the other specialized as well as regional app stores are also available with thousands of apps, which are thrown towards app users. There are two most popular amongst the above-mentioned play stores; one is Google Play store and the other is Apple App store. These apps distributing platforms are termed as built-in applications and these are configured in each mobile as per choice of the mobile manufacturer companies and their developers.

In general, a mobile app extracts content and information from the Internet in the similar fashion to a website; it is also found that number of mobile apps on Play Stores is increasing every year to facilitate a user with different types of applications. In response to the improvement of an app, a feedback is termed to be such thing, which is gathered from app users. By means of feedback received by mobile users, there are two tasks, which are achieved; the rank of apps is measured, whether which app needs improvement, besides it, app requirements are extracted to decide, either to launch a newer version or wait for the next survey. Feedback actually comprises of ratings and comments in the form of user reviews, whereabouts the statistics of the number of downloads of an app is also followed by the app developers to come to know, whether it requires any updates and changes or it is fine enough to be used by users.

Sometimes, app users unintentionally supply such ratings for a kind of their experience or something else, which lead a developer collecting non-rigorous app requirements, and these requirements become a cause of developing non-flexible and uncertain or unusual version of mobile app. Herein-forth, a developer of mobile app launches such newer version on play store by putting such changes, which are actually baseless by just depending over the wrong evaluation of requirements. So also, it becomes more challenging and hectic to a developer for extracting accurate requirements. Hence, a user after experiencing it with bogus changes, s/he removes such app because of

non-flexibility in latest features as per comparison with the features of earlier version.

It needs to be reiterated that during literature review, it is suggested throughout several studies that the concurrent changes in an app should be based on the wish of users, which can be found from reviews made by them in the form of feedback as well as to be followed by mutual and independent policy proposed by this research study. However, the trend of frequent changes in a mobile app is also a kind of headache to a user, which leads to an app removal or un-installation. A developer should keep in his mind that several users are no more habitual of adopting frequent changes because they have completely understood behavior of recent version and newer version destroys their present rhythm.

Even though as per literature, different practices are shown in several research studies with good emerging techniques but fewer of them are succeeded in facing the challenge of extracting optimal requirements with less chances of their failure. However, after confirming the results and potential of such successful studies, some of them are thoroughly reviewed and followed by this proposed research study for modeling such feedback schema. The reason behind selecting ontological schema for the proposed study is to interact with semantic search based strategy, as it helps in evaluating such factors of evaluation, which retrieves valuable information by avoiding other additional entities.

It is generally observed that the conceptualization of a user feedback is missing because of this fact, we include ontological interface and we know that ontology is a way to fill-up such sort of gap, i.e. what sort of information is contained in the user feedback as well as what sort of information can be extracted from it. However, an ontology is an explicit description of the domain of interest in respect of concepts along with relationships among them, besides it, the properties are available and constraints over the properties to make them more specific. Nonetheless, many tools are available for practicing and modeling ontology such as Protégé, NeOn, etc, on the other hand, some of the graphical tools are there to design diagrams i.e. Graphoo, etc.

Whereas, the concept behind suggested study is more elaborated and illustrated in two different aspects; user reviews (i.e. ratings and user comments) and the other is, number of downloads of an app. Generally, most of the recent popular studies follow a trend of choosing the number of mobile app downloads; as if any of the app has the highest number of downloads, it means, the app looks good enough to be utilized more and no changes are required.

The actual story is that many users are habitual of downloading newer apps for experiencing more and more on the smart phones, just because of their experience and practice of downloading apps, several mobile apps are set aside with no changes/updates, it is just because of

keeping in mind that its' features completely satisfy an app user. The fact mentioned in this paragraph is against the trend of frequent changes or updates in mobile apps.

There are many barriers to a mobile app user, which are actually occurred by frequent updates or unusual version updates of a mobile app, i.e. old version of app is better than the newer one. Being a mobile app user, frequent updates are the most vulnerable, which destroy customer rhythm with the version of an app, besides it, if a user finds an unusual version update, as the previous version is better and the newer is less charming, then, in some of the cases, a user uninstalls or leaves the app for making its more use, which negatively impacts over the rank of an app. It is, therefore, the present study focuses on updates with mutual policy i.e. app ratings through auto-reviews, so also keeping it as one of the drawbacks towards the launching newer versions of an app for scrutinizing every time with a major attention of the mobile app developer.

Hence, the actual status or statistics of a mobile app can easily be evaluated and the selected policy can be applied without processing extra parameters and saving time by avoiding extra efforts, which would be taken during comparison of other parameters of an app.

The following pictures show a process of ratings of a mobile app, so also captions of the rating stars are labeled against each level of rating schema.



Fig. 1 A general mobile app rating technique

## User Feedback in Software Engineering:

It is learnt throughout the study that feedback is relevant to the evolution of software, it is also found as one of the phases of lifecycle of software i.e. Software Evolution. Its aspect may also point out new designs from old ones to be created on the basis of some parameters, which are identified in collaboration with the team or sometime from the surveys conducted from the audience. The user of a software sometimes need changes in the software according to the required tasks to be executed in an organization, whereas it is all about meant to be that part of a software for fulfilling required targets.

Madhavji et al. (2006) defines feedback as, "It is one of the most primary results of introducing implemented software systems in real world. There is actually a prompt response to the system from those affected by it." However, the explicit user feedback can be recognized from it that it leads both as an artifact and as a process. Whereabouts, a user feedback is a kind of reaction made by a user upon his experience in using a software, for either liking or disliking with its existing features, not only this but also containing meaningful information, in which a user suggests for improvements by pointing failures of such software. Besides it, a user feedback is one of the communication processes, which is made between the provider and user of software.

If we are looking the feedback in requirements discovery, it is stated "effective feedback is being received at the time when somebody understands something very well enough to be able to constructively react to it" (Madhavji et al. 2006). On the other hand it can be defined that some of the users give feedback unintentionally and they do not even understand the mechanism of software and leave their feedback without pursuing it properly, which lead a software developer towards the non-flexible version to be launched. It leads us towards the first two phases out of four phases of the actual feedback cycle, namely, feedback collection and feedback analysis (Brun et al., 2009). The collection of data with regard to the feedback is necessary for evaluating the statistics of software, and it can be collected from the repository, which is dedicated to gather users' feedback. Whereas, apart from collection, feedback analysis is also the part of feedback cycle, in which analysis of comments is carried out, nevertheless, there are several approaches which apply Natural Language Processing for the purpose of analysis of a feedback. The speed of the feedback analysis is sometimes dependent on the way of collecting feedback. What do we mean of it? is that for example there are some platforms which collect the feedback by letting a user classifying it as feature request, bug report, enhancement, etc. Whereas, there are some approaches which analyze only one class of feedback, such as feature request (Cleland-Huang et al., 2009), or that analyzing feedback to determine its different

classes as bugs, features or clarifications (Carreño & Winbladh, 2013). Another one important difference in between implicit and explicit feedbacks is that the latter moves from the user intention for communicating someone something on the basis of their beliefs and goals so also sometimes move the audience to act consequently (Itzel Morales-Ramirez et al, 2015).

Table 1: Properties of a User Feedback

Basic properties (input)	Analysis technique	Derived properties (output)
Title	POS tagging Topic modeling Sentiment analysis	Features Topics Sentiment
Description	Same as for Title+ Speech act annotation Sentence parser	Same as for Title+ Intentions Deontic moods Length

User feedback properties are presented through title and description, as shown in table 1, wherein analysis techniques and properties are entangled with both properties of feedback.

### General mobile apps analysis technique:

- App rating;
  - Rating = Stars (#1 throughout #5)
    - Hated, ii)Disliked, iii)Its ok, iv)Liked and v)Loved it, respectively
- Users' comments;
  - Sample Comments = n (desired number)
    - Only a sample of reviews is analyzed through experts opinion without queuing the remaining data in process
    - Some of emerging techniques have recently got much scope to cope with such drawback, e.g. NLP(Natural Language Processing) and Speech Act Theory(SAT)
- Number of downloads;
  - Success point = h (highest no. of downloads)
    - It is also found that a mobile app iteratively downloaded from app store gets more success against the apps downloaded rarely

Aforementioned variables are the indicators of success point of a mobile app. However, a user of an app and its' developer are the only influencers of the study, which are found so far in the literature. However, a developer is concerned with the releasing perspective of a mobile app, whereabouts a user is to utilize it for the purpose of fulfilling desired tasks.

### Variables affecting market feedback:

- Independent rating system
  - App ratings with autonomous schema to analyze such data, which is supplied by app

users whether for the sake of newer version or for removal of any sort of bug from mobile app

- *It is the only fact to pick ontological schema for the proposed research study because an ontology is the best source of encapsulating independency by applying desired constraints over the properties to make them more specific and meaningful*

## 2. Frequently updates

- Mobile app updates, which are other than periodic updates or without surveying recent market of such apps

## 3. Unusual version updates

- Earlier version was better than newer version or something missed in newer one

## 4. Need analysis for updates

- A market survey is required to be conducted by the developer of mobile app before putting an app in the queue for updates

## 5. Comparison analysis

- Versions i.e. earlier and newer must be compared before throwing towards recent market, in the essence of app analysis, initially, a developer should conclude analysis

## 6. Customer satisfaction

- It is generally known to every developer that a Customer is the only key to success an app, for which an app should be articulated in view of customers' satisfaction rather than his own putting technicalities

## 7. User oriented terms

- No bombastic words should be used in interface, which a user cannot understand very well. Only user oriented terms should always be followed

## 8. Customer collaboration (agile manifesto)

- Collaboration with customers of a mobile app by means of online feedback for the purpose of facilitating them more and more

The market variables, which are mentioned and shortly defined hereinabove have got high impact over the feedback mechanism through several research studies, whereas among such variables, Sr.No.1 is all about autonomous mobile app ratings as already discussed briefly, which can be presented by applying such proposed ontological schema, while Sr.No.2 to 4 indicate policies of updating mobile apps, which are adopted by mobile app

developers during development of apps, sometimes these update policies become much crucial that a developer goes through the deep analysis technique to cope with such circumstances, whereabouts Sr.No.5 to 8 present those parameters, which are beneficial to come to know about its' customers and their level of satisfaction with the app. It is observed that Agile manifesto i.e. Sr.No.8, is receiving good support from the users just because a customer is given small stories within a short interval of time for their experience. Nevertheless, these variables lead a developer in developing a good mobile app without other consequences. The proposed study is adhering on such variables; the independent rating system is followed throughout the research study to facilitate a developer in acquiring such requirements from the perspectives of mobile app users.

## Applications of the ontology

Scenario-1: Unstructured user feedback

Scenario-2: Semi-structured user feedback

Scenario-3: Structured user feedback

Scenario-4: Semi-automated analysis of user feedback

## DRAWBACK: Perspectives of app ratings and reviews

In literature review, most of relevant studies follow a method of mobile app ranking for whether launching newer versions by extracting requirements or just going through level check of app by exploiting app ratings and user reviews; whereas, certain method followed by studies is as;

### App ratings:

The mostly seen traditional and famous user app feedback mechanism is hitting Star-1 throughout 5, representing Hate, Disliked, Its ok, Liked and the Loved it, respectively. Whereas, users of these mobile apps are separated by the number of ratings i.e.  $10/6=3(\text{avg.})$ , etc.

As per method, value of rate increases by selecting Star-3, whereabouts, Stars-1st and 2nd are automatically highlighted and the ratings are collectively analyzed by their weighted level without focusing on other additional parameters, so it fluctuates requirements of an app and does not provide correct aspect of requirements for either to launch updated version or to wait for a next survey. In a few studies, they keep 10 points on each app by neglecting the method of five concurrent stars, to follow their own mechanism and to scrutinize the app ratings process. A mobile app developer, quantitatively focus on mechanism rather than comparing the app versions.

### User comments:

The reviews of mobile users i.e. comments are gathered and analyzed by the experts and decision makers through their expertise to come out with correct solutions for either launching the newer version or to check the rank/quality of

an app. It is also analyzed that a single app user cannot have more than one feedback for an app as well as the study can implement a limited number of reviews for an app, which are collected as a sample for the said study/app and forwarded to the concerned development team for further scrutiny i.e. extracting requirement, etc. The comments take much more time to decide the statistics of a mobile app; only because of this fact, the proposed study recommends the mechanism of auto-reviews through the ratings i.e. concurrent stars, which actually fluctuates the exact statistics.

Besides both of the elaborated drawbacks discussed hereinabove, another one is that the frequent and unusual updates destroy the rhythm of a user with the mobile app, because of it, rarely a user would like to use after update by seeking its newer features but most of the users neglect to use the same app again because of less charming. The study focuses on the core idea of decision making by collecting requirements accurately and going through the analysis of an app by means of ontological interface.

Towards proposed schema, the following research questions are identified and discussed in the methodology for more purifying such ontological model;

#### Research Questions:

1. Extracting requirements in perspectives of feedback
2. Measuring quality of a mobile app for deciding whether to launch newly featured version or not
3. Designing Ontology based model
4. To defend the questions cited hereinabove, the research study proposed an ontological schema, which is structured through Protégé, Graphoo and an ORSD document, which are explained in the section of methodology, as below;

## 2. Methodology

#### Ontological model based approach:

As it is discussed earlier that the proposed research study presents a model based approach via an ontological perspective to define and demonstrate the logical connection in between ratings and comments, which can easily be understood by a system for the purpose of producing optimal results in favor of abstract, which is mentioned hereinabove. It is well known that the reviews are scrutinized by means of ratings and comments i.e. feedback. Whereas, the ratings are followed by stars i.e. 1 throughout 5 and the comments are analyzed through the experts' opinion and decision makers.

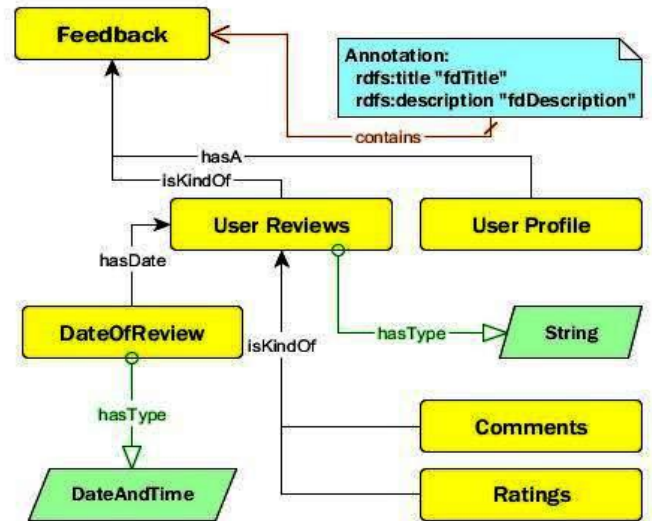


Fig. 2 Presentation of general Feedback schema

A simple feedback comprises of i) user profile, ii) user reviews e.g. ratings, comments and iii) date, as it is presented hereinabove in Fig.1, which is usually followed and suggested by several relevant studies in their proposed work as well as future directions. Whereabouts, a single user review is dropped down into two sections of Ratings and Comments; a rating consists of stars; 1 throughout 5, which are presenting Hated, Disliked, Its ok, Liked and loved it, respectively. On the other side of research study, comments are comprised of user written reviews in specified sort of language, whereas, the date of review keeps date and time, on which, review by the app user was made.

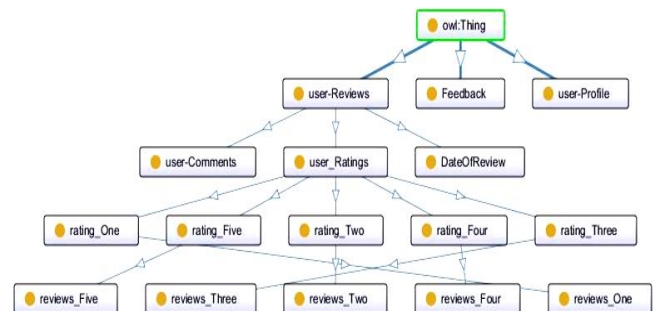


Fig. 3 Onto-Graph, presenting hierarchical relationship between concepts

It is diagrammatically presented herein below by putting the different concepts i.e. classes in a protégé interface, with regard to the ontology development, which are shown in a hierarchical relationship in Fig.2, through which their logical phenomenon is arranged.



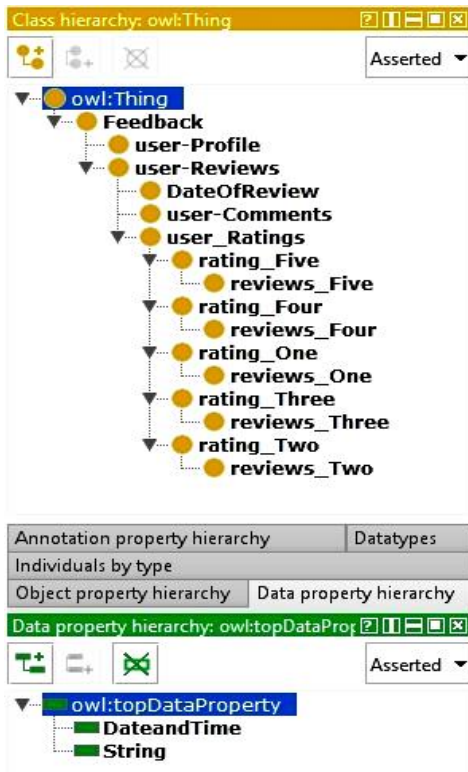


Fig. 4 Class hierarchy and data property hierarchy

It is a proposed sketch of ontology development, as presented hereinabove in Fig.3, it consists upon different classes i.e. Feedback, whereabouts, user\_Profile and user\_Reviews are shown as its child classes, whereas, user\_Reviews is more divided into DateOfReview, user\_Comments and user\_Ratings, so also, a user\_Rating is dropped down in more child classes. However, only two of the data types are used; DateandTime and String, for the purpose of encapsulating data in a repository, accordingly. In the essence of suggested ontology schema, initially, concepts, properties and their relationships are configured in a protégé environment, shown in Fig.3; as the same is mentioned earlier, afterwards, specified data types are set to gather such data in the concerned set of repository. Thereafter, their usage can easily be seen in Fig.4, which is also presented herein below, in which, it is collaboratively enlisted that to which a property is disjoint with and its domain/range along with its sub-property.

Moreover, in view of expanding more aspects with regard to the proposed ontology development via protégé tool; the designed sketch of data properties and usage of such concepts is presented and elaborated, as under;



Fig. 5 Data property and description to concepts

In the essence of a general programming paradigm, while rendering its concepts, it is learnt that a data property is something in a domain of concept, which retrieves value of a data attribute of an object in between such repository and the interface provided to a user.

Data properties are tightly coupled with the objects of such concepts to encapsulate their desired data type with their specified module; accordingly, relevant data is easily collected and stored in suggested repository. There are two data types, which are picked in recent study i.e. String and DateandTime; String for propagating the concept of comments, which are actually submitted by app users in the form of English written lines.

It is also observed while rendering and doing practice over different objects in such interface, which is provided by protégé tool, these concepts, data properties and all other items mentioned in a protégé tool need to set their domain and range for providing their direct connection with proposed domain, so that data retrieval is made from module to module.

As per expression defined in the module of ontology i.e. 'DateandTime DisjointWith String', it means, DateandTime is set as a data property in the domain, which is disjoint in terms of data retrieval with the other data property i.e. String, as shown in recent study. It becomes more clarified by using the term disjoint in an expression of ontology development that an object or its attribute does not mingle with the other specified object or its attribute in a domain throughout the whole structure of a model.

In view of the proposed research study, the comprehensive ranking of the mobile apps against the above cited drawback is discussed, as below;

Initially, the app ratings and comments are interleaved to each other by mutual policy, which is easily designed and implemented in an ontological interface by mapping the entities of study with different aspects with the help of using schema, which is provided by Protégé-an interface for designing and implementing ontology.

Secondly, the comments are covered in five aspects of requirements so that these can be analyzed and processed by a system in a suggested ontological interface. Whereas, these five aspects are shown around the comments to make them more specific and meaningful as well as machine understandable with ease of use.

Table 2: Ontology Requirement Specification Document (ORSD)

1	<b>Purpose:</b> a) The aim of the proposed ontology schema is to provide a comprehensive ranking method of the mobile apps with good results in comparison to recently used schemas and techniques b) The apps are ranked through user feedback (comprises of ratings and comments) c) Sometimes, the ratings are made by the users unintentionally, which lead a developer towards non-rigorous requirements and in a result, it causes non-flexible app versions to be launched over the app stores d) The ratings are collectively analyzed by their weighted level, so it may fluctuate the potential of requirements of an app e) An ontology for the domain study is the best way to extract requirements with less challenges
2	<b>Scope:</b> a) A single user cannot have more than one feedback for an app, which is supplied; after making a user profile or it can also be supplied without making a profile b) The study can implement a limited number of reviews for a mobile app, whereas, an unlimited number of reviews is crucial to decide its statistics because it will take long time for scrutinizing
3	<b>Level of formality:</b> The ontology is to be developed in Protégé i.e. OWL language The graphical design of study is to be mapped through UML representation tool i.e. Graphoo
4	<b>Intended users:</b> Smart phones; Android and Symbian Mobile app users
5	<b>Indented uses:</b> a) Providing accurate ranking of the apps b) Requirements extraction which may lead to launch a newer version of the app with latest features c) Interpreting an ontological schema d) Representing UML design
6	<b>Group of competency questions:</b> a) What is the logic behind the proposed study? i) The suggestions are analyzed by experts and decision makers ii) The rating depends over stars, which are of 1 throughout 5
7	<b>Pre-Glossary of terms:</b> a) Mobile app, feedback analysis, ranking b) Mobile phones and users

The ORSD document helped out in partitioning the recent study factors, which are to be utilized in ontology, which

meant to be the part of proposed study for acquiring results and their implementation.

### 3. Future Work

An interface of multi-criteria is still required to be structured and formulated in the perspectives of ontological interface, which can easily be dealt with the unlimited or infinite number of app users and their feedback gathered in a repository, while in most of the cases, a sample of users feedback is usually taken by selecting a limited or finite number of reviews. Nevertheless, at present, it is a difficult task to achieve the suggested goal with just an ontological operational interface; it requires collaboratively use of data mining and ontological perspectives. It is, therefore, suggested hereinforth to have a strong tool support of an interface providing multi-criteria techniques, for the purpose of enumerating best results and to add more quality to the proposed mobile app rating schema in recent studies. Whereabouts, an interface comprising of multi-criteria techniques could be offered by putting dynamic analysis schema to retrieve such results within small interval of time. Besides the above suggested multi-criteria technique, the mobile app ratings schema should be interleaved with NLP (Natural Language Processing) or SAT (Speech Act Theory), which can easily be encapsulated in different ontological perspectives by looking in to the user reviews i.e. comments supplied by mobile app users in English written lines. Ontology engineering being a field of software engineering is one of the techniques of semantic web search, which encompasses concepts, relationships among such concepts and properties over the concepts along with constraints over properties, for making them more and more specific.

### 4. Conclusion

The research study with regard to the proposed feedback ontology, is thoroughly conducted to scrutinize usual mobile app ratings schema, which is mostly adopted by the recent market developers for the purpose of performing analysis of ratings technique, henceforth, the proposed study is followed by software engineering lifecycle via modeling ontological schema i.e. requirements provided by app users are gathered in terms of feedback, later on the proposed an analysis technique, which is used to encompass the retrieval method of requirements extraction, afterwards specified ontological parameters are set according to the ontological interleaved schema policy, whereabouts in the end a mature format of feedback is extracted from the schema, which is not only easy for practitioners to develop a newer version, so also to launch a version of an app with fresh updates. In short, proposed

schema i.e. ontological feedback analysis, is structured through the use of Protégé; ontology development tool and Graphoo; a UML (Unified Modeling Language) representation of concepts, properties of concepts and relationships among them. In some of the special cases, these properties of concepts are covered by putting some constraints i.e. axioms to make them more and more specific, so also an ORSD; a document to elaborate concepts, entities and tasks.

## References

- [1] Itzel Morales-Ramirez et al, 2015, "An ontology of online user feedback in software engineering," Applied ontology 10(2015) pp. 297-230
- [2] Wajid et al, 2019, "Reasoning and representation of legal cases through ontology", IJCSNS, Vol.5, No.5, September, 2019
- [3] Itzel Morales-Ramirez et al, "Exploiting user feedback in tool-supported multi-criteria requirements prioritization,"
- [4] Li Li et al, 2016, "Static analysis of android apps: A systematic literature review,"
- [5] Dennis Pagano, Walid Maalej, "User feedback in the Appstore: An empirical study,"
- [6] Stuart McIlroy, Nasir Ali, Ahmed E. Hassan, "Fresh apps: An empirical study of frequently-updated mobile apps in the Google play store,"
- [7] Venkata N Inukollu et al, 2014, "Factors influencing quality of mobile apps: Role of mobile app development life cycle", IJSEA, Vol.5, No.5, September 2014
- [8] Hans-Jorg Happel and Stefen Seedorf, "Applications of Ontologies in Software Engineering"
- [9] Maleknaz Nayebe et al, "App store mining is not enough", 2017 IEEE/ACM 39th IEEE International Conference on Software Engineering Companion
- [10] William Martin et al, "A survey of app store analysis for Software Engineering", 2016
- [11] Gai Li et al, "Exploiting explicit and implicit feedback for personalized ranking", Volume 2016



**Qamar Uddin Khand** received the PhD degree in Production and Information Systems Engineering from Muroran Institute of Technology, Hokkaido, JAPAN in 2005. He is currently an Associate Professor & Program Director of School of Animation, Department of Computer Science, Sukkur Institute of Business Administration University (Sukkur IBA University), Sukkur, Pakistan. His current research interests include Embedded Control Systems, Sketch-Based Interfaces and Modeling, Soft Computing, Computer Graphics and Animation, and Software Engineering.



**Sajid Ahmed Ghanghro** received degree of Masters of Science (Software Engineering) from Sukkur IBA University, Sukkur Sindh, Pakistan in the year 2019. His research domain is Software Engineering, Semantic Web and Human Factors in Software Engineering, while he has recently worked on software engineering ontologies. He has more than three years of experience in Teaching and Research work.



**Wajid Ahmed Channa** received degree of Masters of Science (Software Engineering) from Sukkur IBA University, Sukkur Sindh, Pakistan in the year 2017. His research domain is Software Engineering and its aspects in other fields of relevant studies, so also semantic web, while he has recently worked on software engineering ontologies. He has more than five years of experience in Judicial Department Govt. of Sindh.