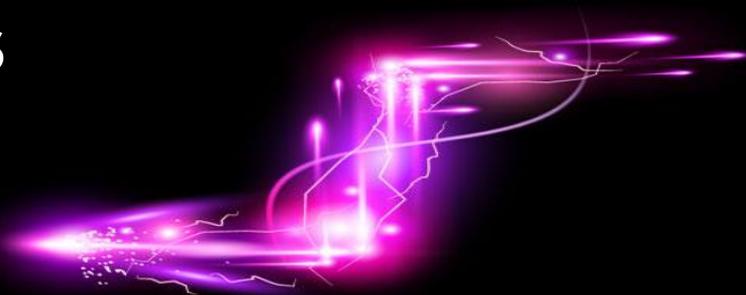


Sentiment Embeddings with Applications to Sentiment Analysis

(Data mining)



SHAIK NAGABABU
MCA 3rd YEAR
1009-14-862-024

Contents



- Introduction
- Abstract
- Existing system
- Proposed system
- Implementation
 - Modules
- Software & hardware requirements
- System design
 - Architecture
 - UML diagrams
- Database design
 - Er-diagrams
 - Tables
- Conclusion



Introduction to Sentiment Analysis

Overview



- What is sentiment analysis (SA)?
- Why is it worth doing?

What is Sentiment?



- Sentiment = feelings
 - Attitudes
 - Emotions
 - Opinions
- Subjective impressions, not facts
- For/against, like/dislike, good/bad, etc.

What is Sentiment Analysis?



- Using NLP, statistics, or machine learning methods to extract, identify, or otherwise characterize the sentiment content of a text unit
- Sometimes referred to as opinion mining

Questions SA might ask



- Is this product review positive or negative?
- Is this customer email satisfied or dissatisfied?
- Based on a sample of tweets, how are people responding to this ad campaign/product release/news item?

Abstract



- We propose learning sentiment-specific word embeddings dubbed sentiment embeddings in this paper. Existing word embedding learning algorithms typically only use the contexts of words but ignore the sentiment of texts. It is problematic for sentiment analysis because the words with similar contexts but opposite sentiment polarity, such as good and bad, are mapped to neighboring word vectors.
- We address this issue by encoding sentiment information of texts sentences and words together with contexts of words in sentiment embeddings. By combining context and sentiment level evidences, the nearest neighbors in sentiment embedding space are semantically similar and it favors words with the same sentiment polarity.
- In order to learn sentiment embeddings effectively, we develop a number of neural networks with tailoring loss functions, and collect massive texts automatically with sentiment signals like emoticons as the training data. Sentiment embeddings can be naturally used as word features for a variety of sentiment analysis tasks without feature engineering.

EXISTING SYSTEM



- ❖ Existing embedding learning algorithms are mostly based on the distributional hypothesis, which states that words in similar contexts have similar meanings. Many matrix factorization methods can be viewed as modeling word representations.
- ❖ We query neighboring sentiment words in existing sentiment lexicon to investigate whether sentiment embeddings are helpful in discovering similarities between sentiment words.

PROPOSED SYSTEM



- ❖ The proposed methods differ from context-based models in that we capture sentiment information of texts, which provides crucial evidences for capturing similarities between sentiment words.
- ❖ The hybrid models can be viewed as the “joint” version of ReEmbed by simultaneously encoding contexts of words and sentiment of sentences into word representation from scratch. Two hybrid models yields best performances as they capture not only contexts of words but also sentiment information of sentences.
- ❖ The approach proposed in this work cannot handle the words not covered in the embedding vocabulary. How to deal with the newly generated words in social media is an interesting future work.

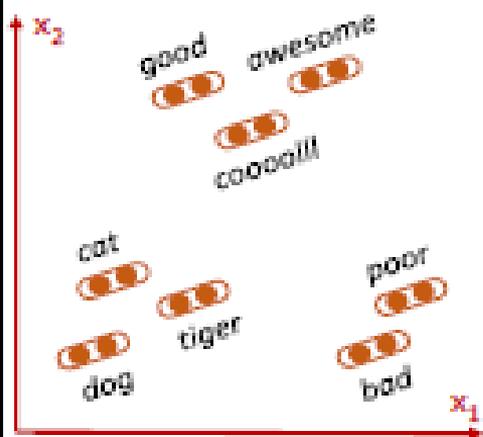
IMPLEMENTATION



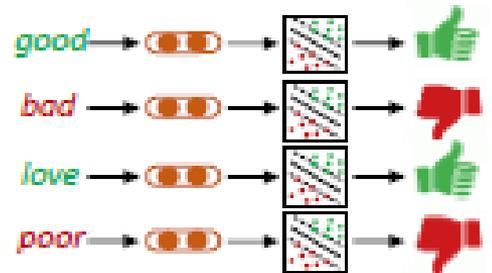
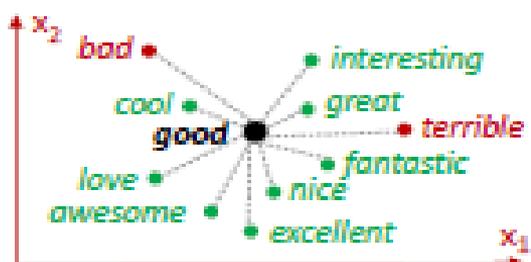
MODULES

- Word Level Analysis
- Sentence Level Analysis
- Building Sentiment Lexicon

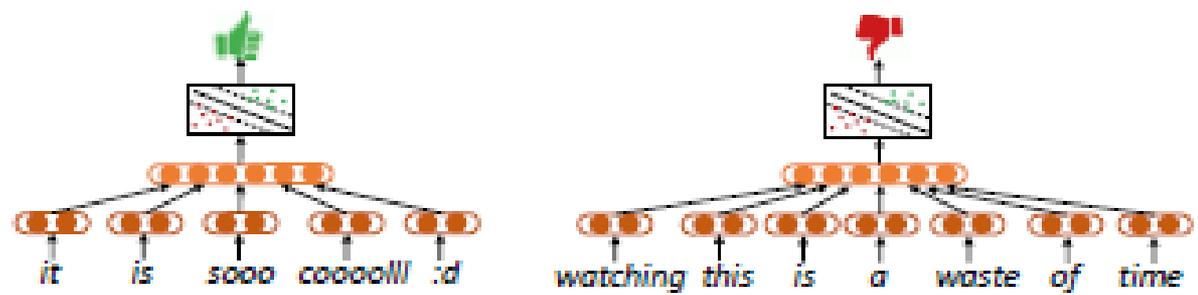
Sentiment Embedding



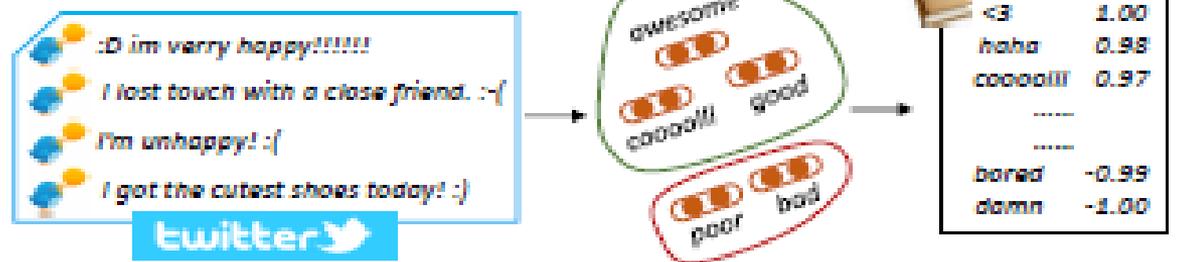
Word Level Sentiment Analysis



Sentence Level Sentiment Classification



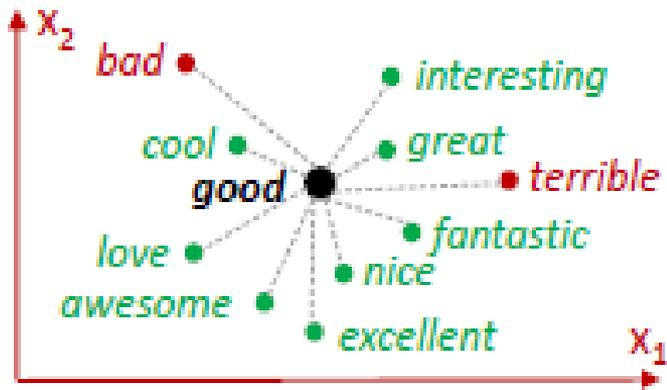
Building Sentiment Lexicon



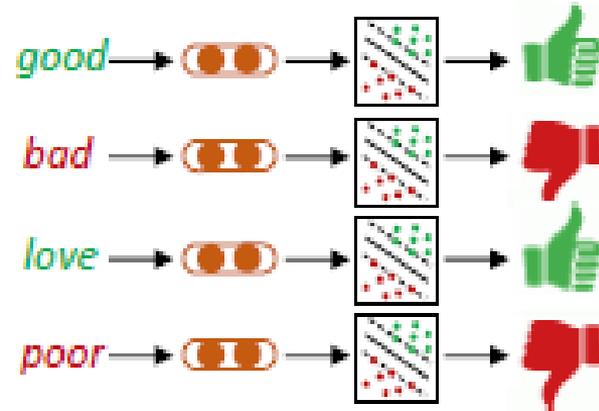
WORD LEVEL ANALYSIS



- We investigate whether sentiment embeddings are useful for discovering similarities between sentiment words in this section.



(a) Querying Sentiment Words



(b) Word Level Sentiment Classification

SENTENCE LEVEL SENTIMENT CLASSIFICATION



- ❖ we apply sentiment embedding as features to sentiment level sentiment classification. This helps us to investigate whether sentiment embedding is capable of capturing discriminative features for classifying the polarity labels (e.g. thumbs up or thumbs down) of text. We first present our strategy of using sentiment embedding as features for sentiment classification. We then describe experimental settings and empirical results

BUILDING SENTIMENT LEXICON



- We apply sentiment embeddings to building sentiment lexicon, which is useful for measuring the extent to which sentiment embeddings improve lexical level tasks that need to find similarities between words.

Lexicon-based

I had **nightmares** all night long last night :(



Text Processing
Algorithm

Negative

System Configuration



H/W System Configuration:

- Processor - Intel i5
- Speed - 2.7 Ghz
- RAM - 8 GB
- Hard Disk - 128 GB SSD

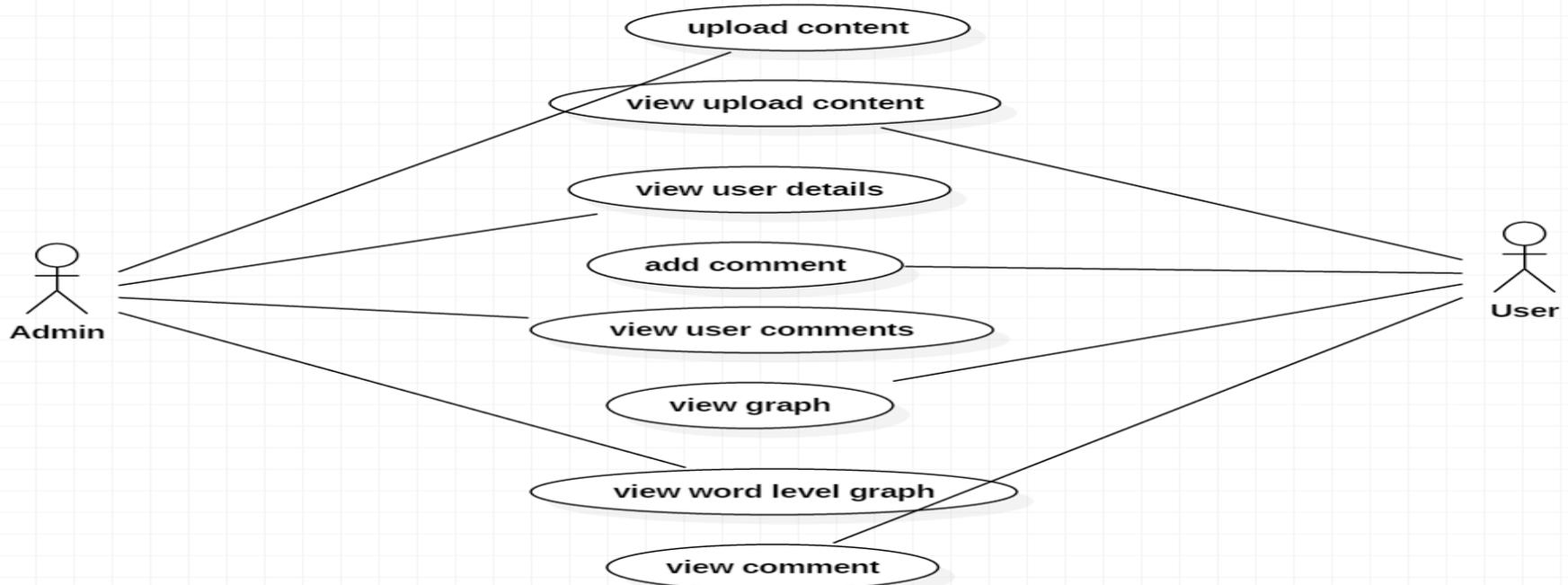
S/W System Configuration



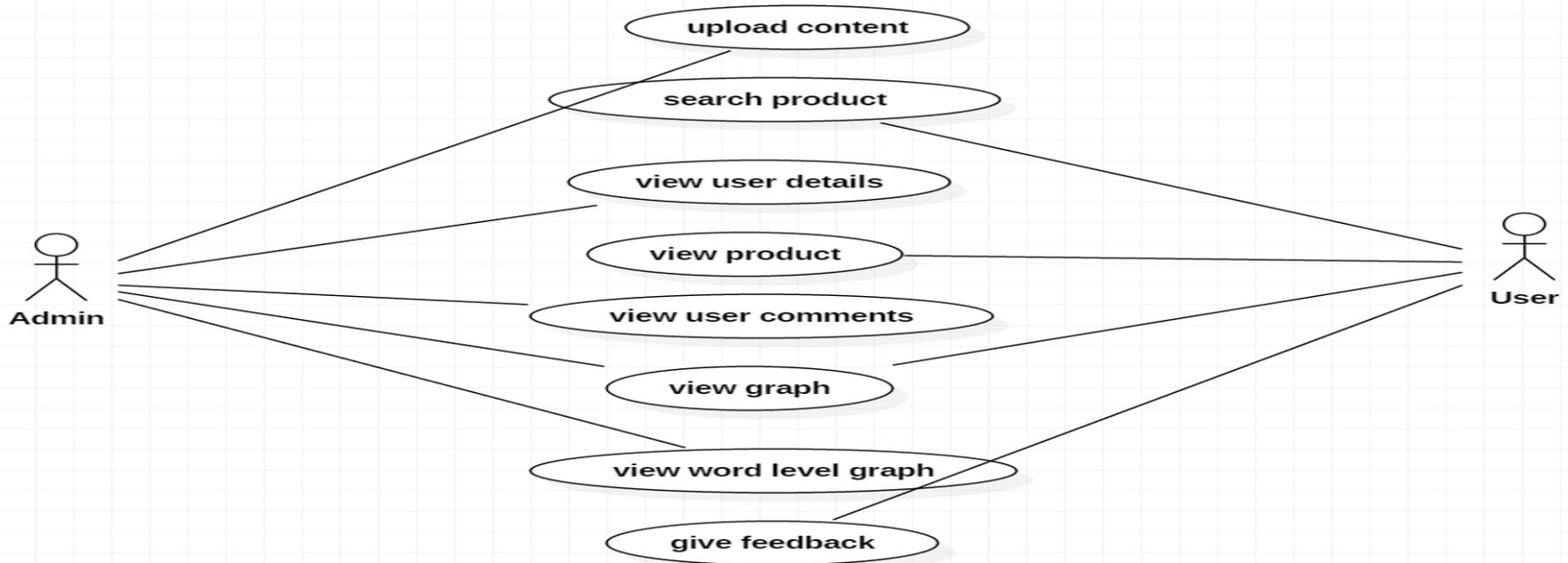
- Operating System : Windows 8.1
- Application Server : Tomcat 8.5
- Front End : HTML, Java, Jsp
- Scripts : JavaScript
- Server side Script : Java Server Pages
- Database : Mysql
- Database Connectivity : JDBC

Usecase diagram

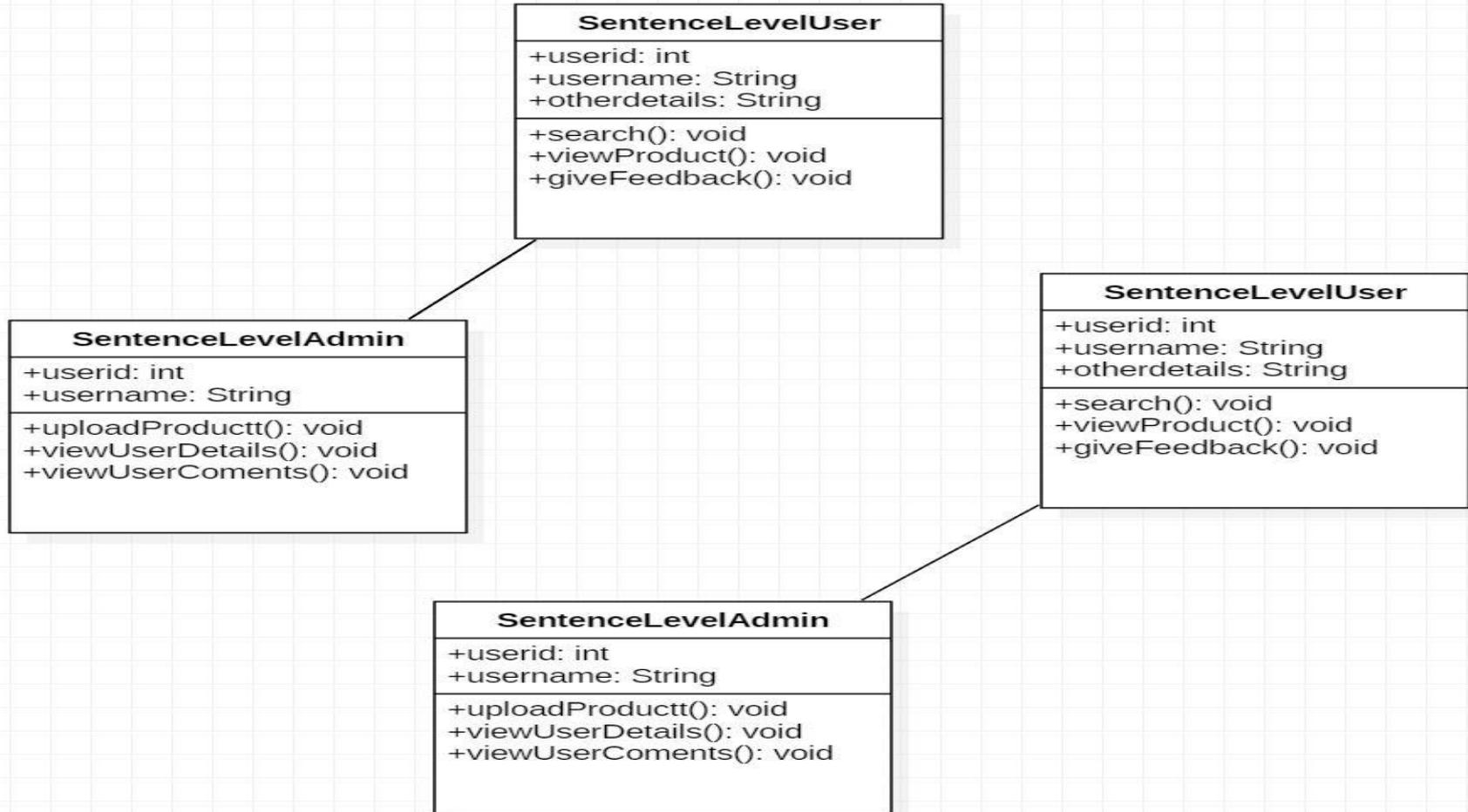
Word level



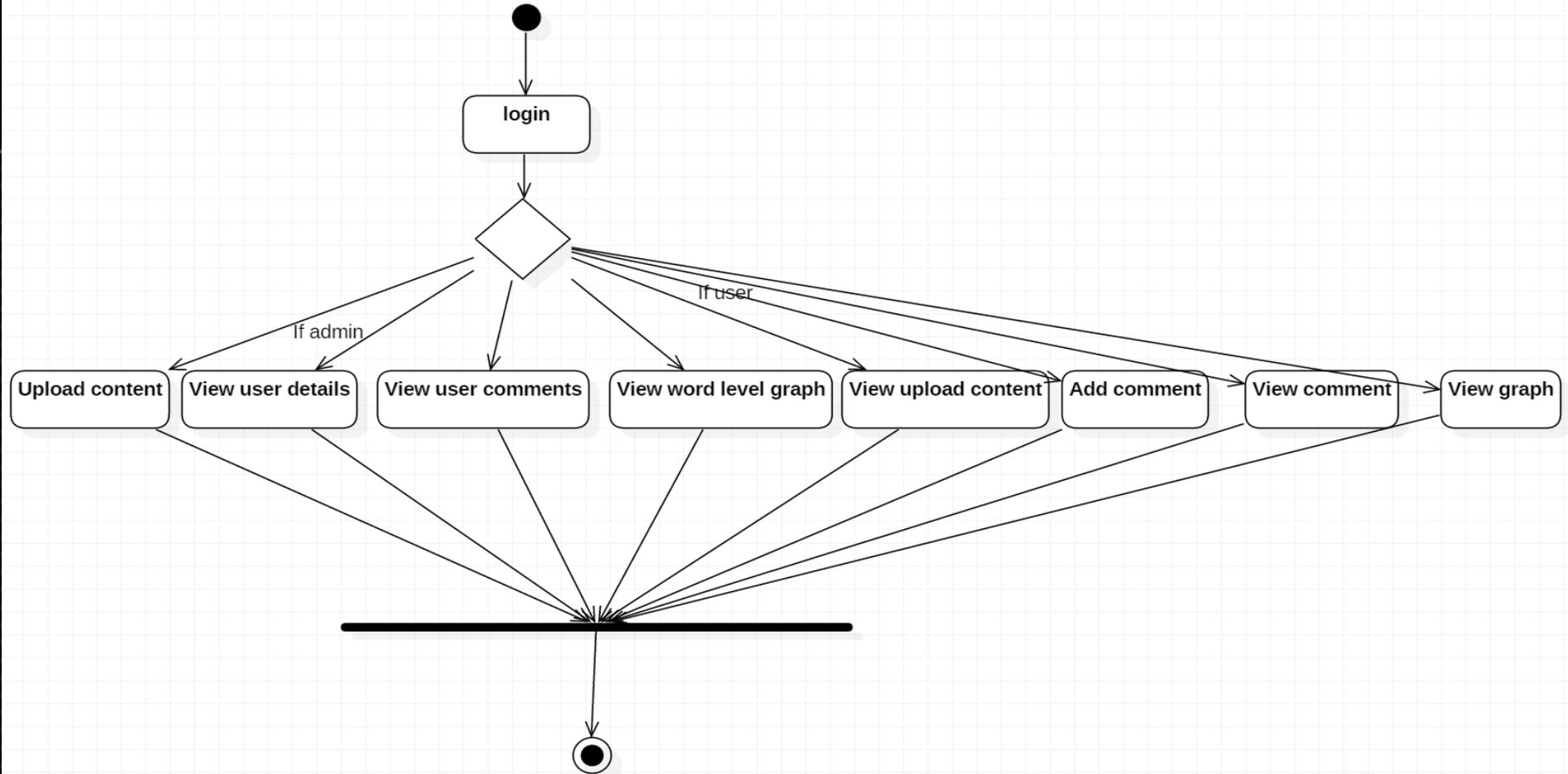
Usecase diagram sentence level



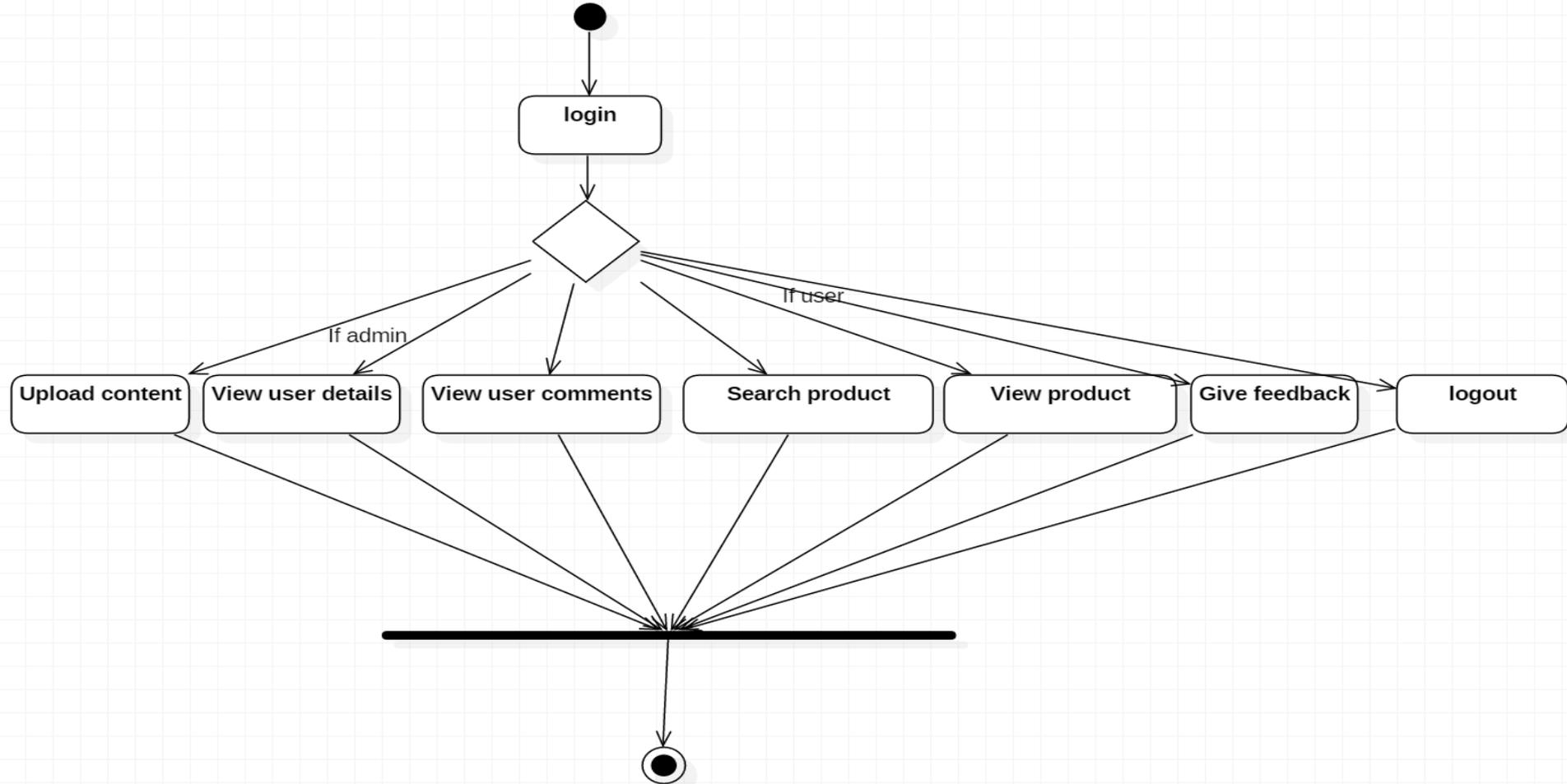
Class diagram



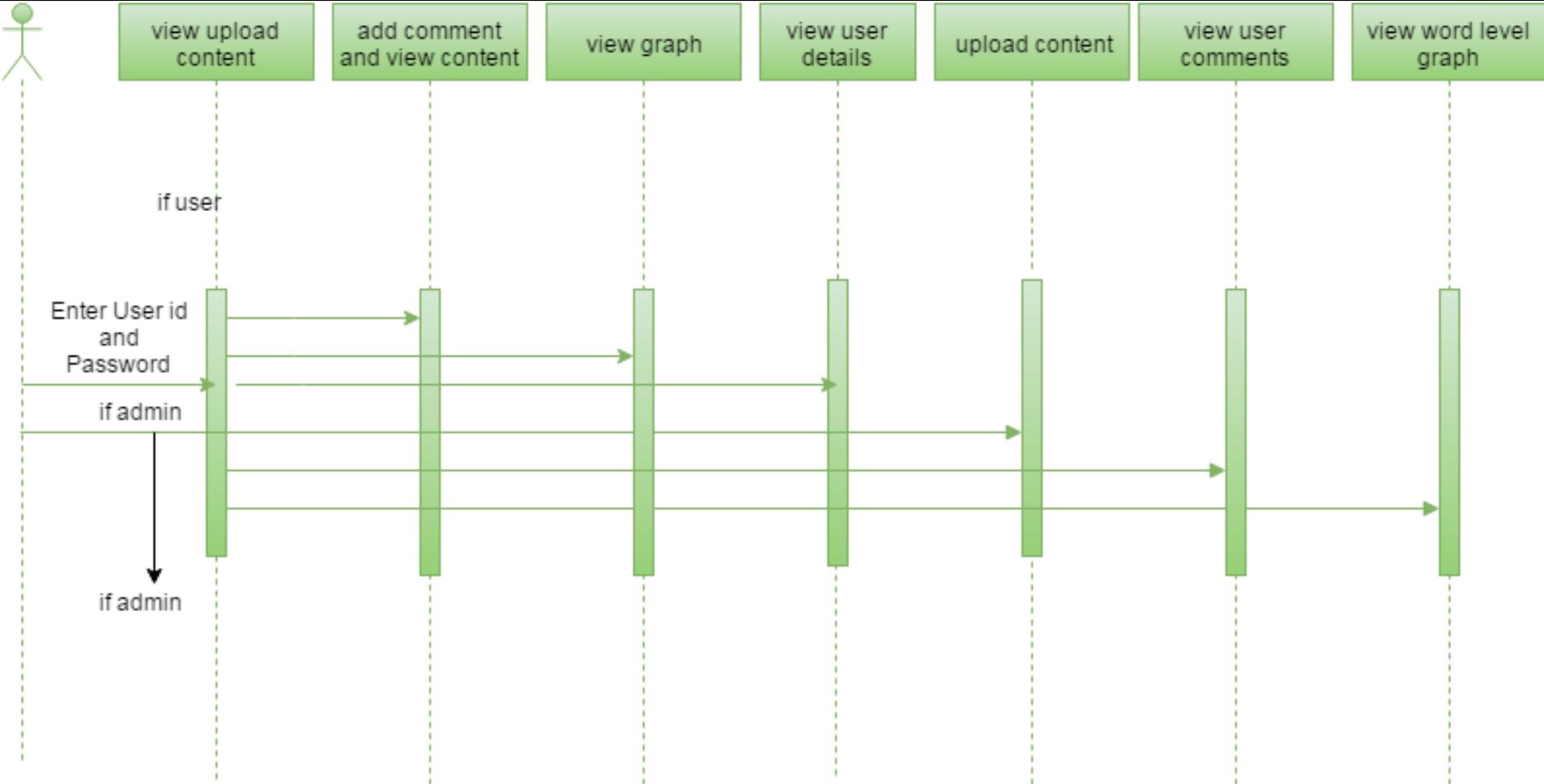
Activity diagram word level



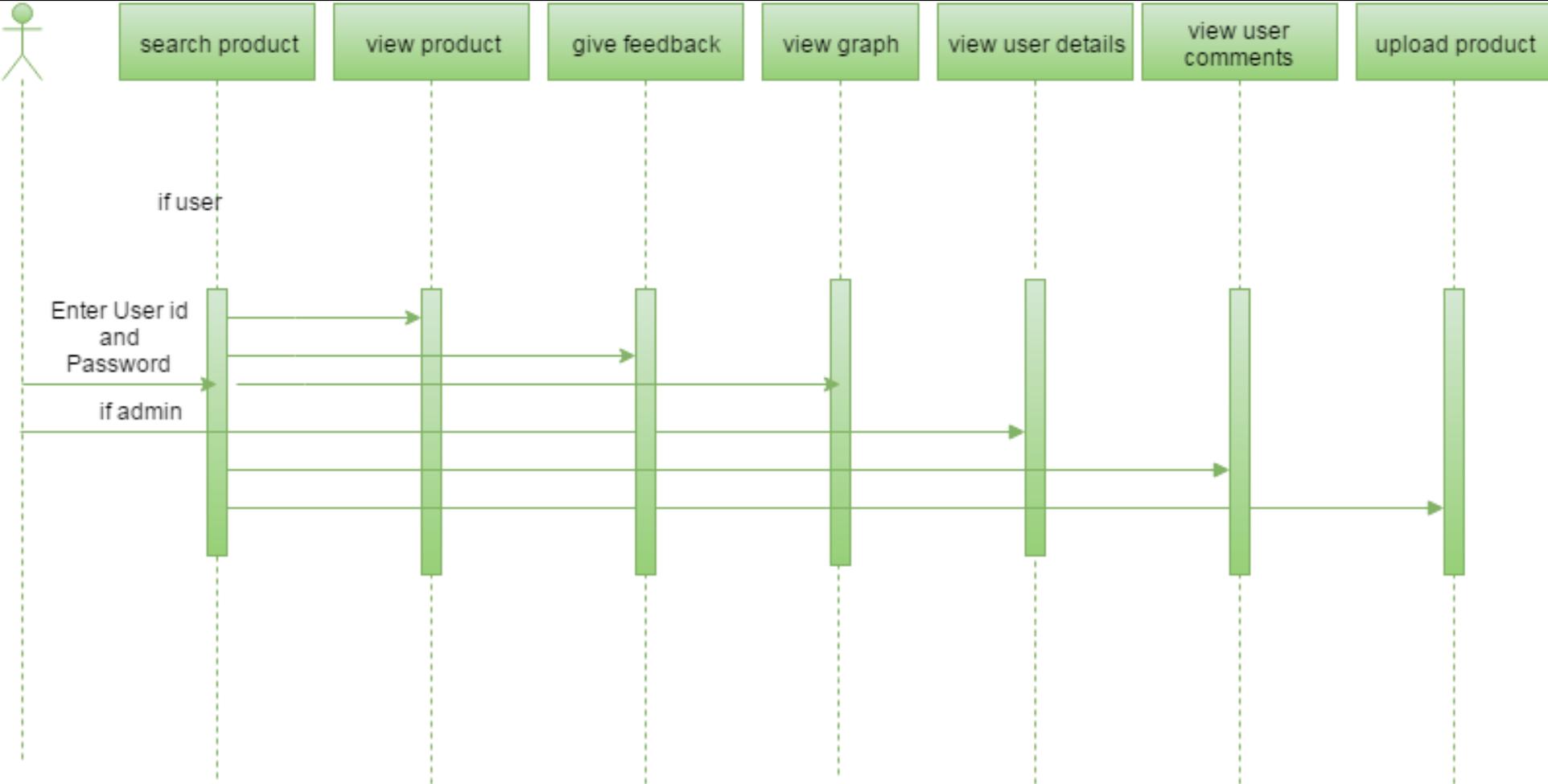
Activity diagram sentence level



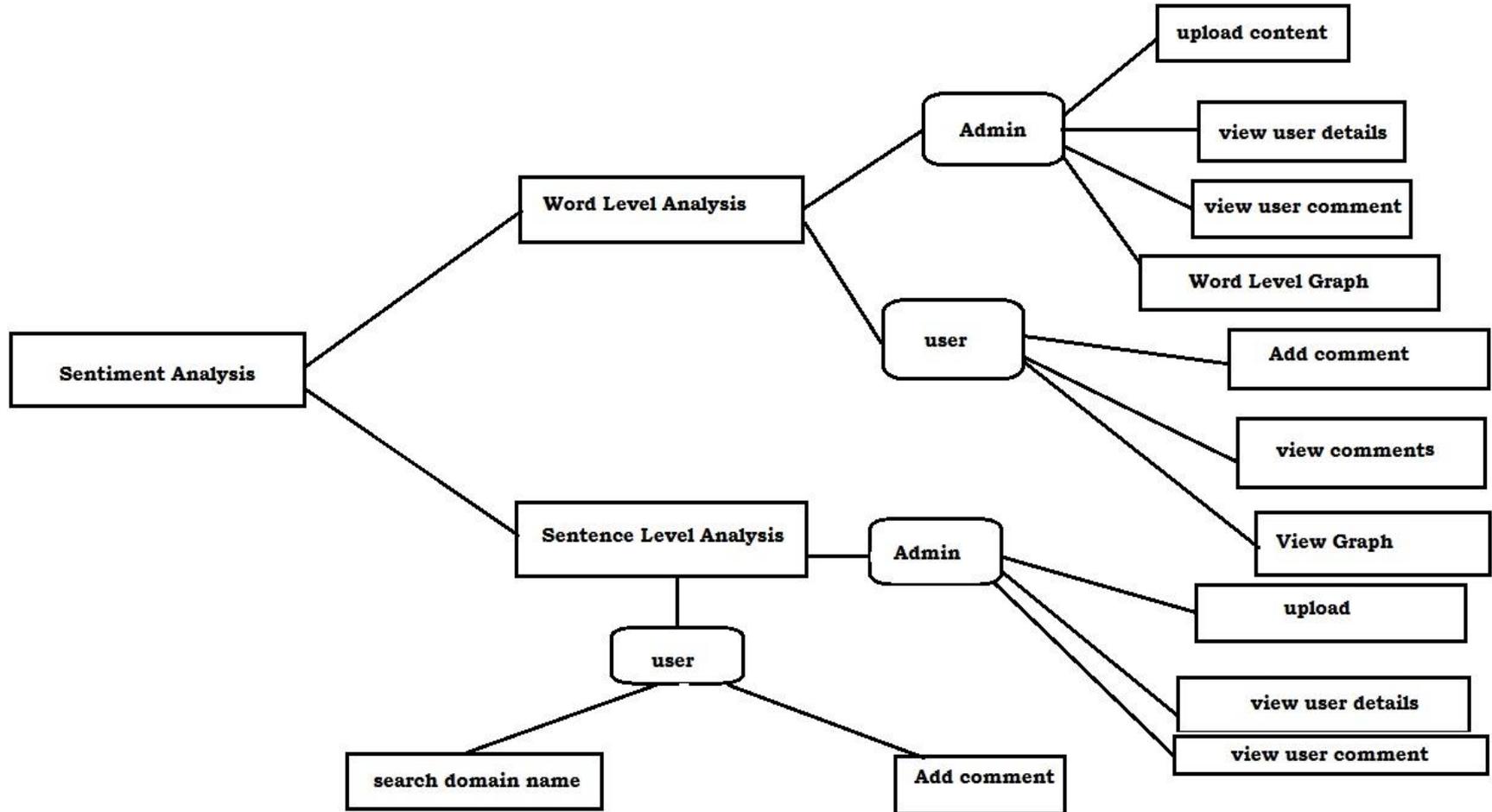
Sequence diagram word level



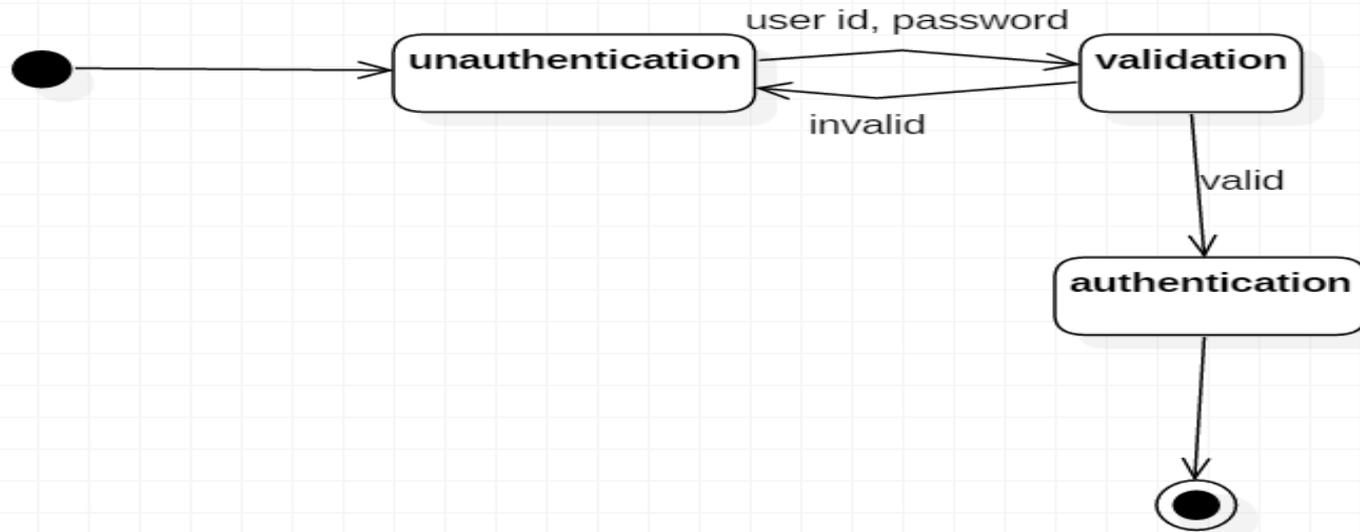
Sequence diagram sentence level



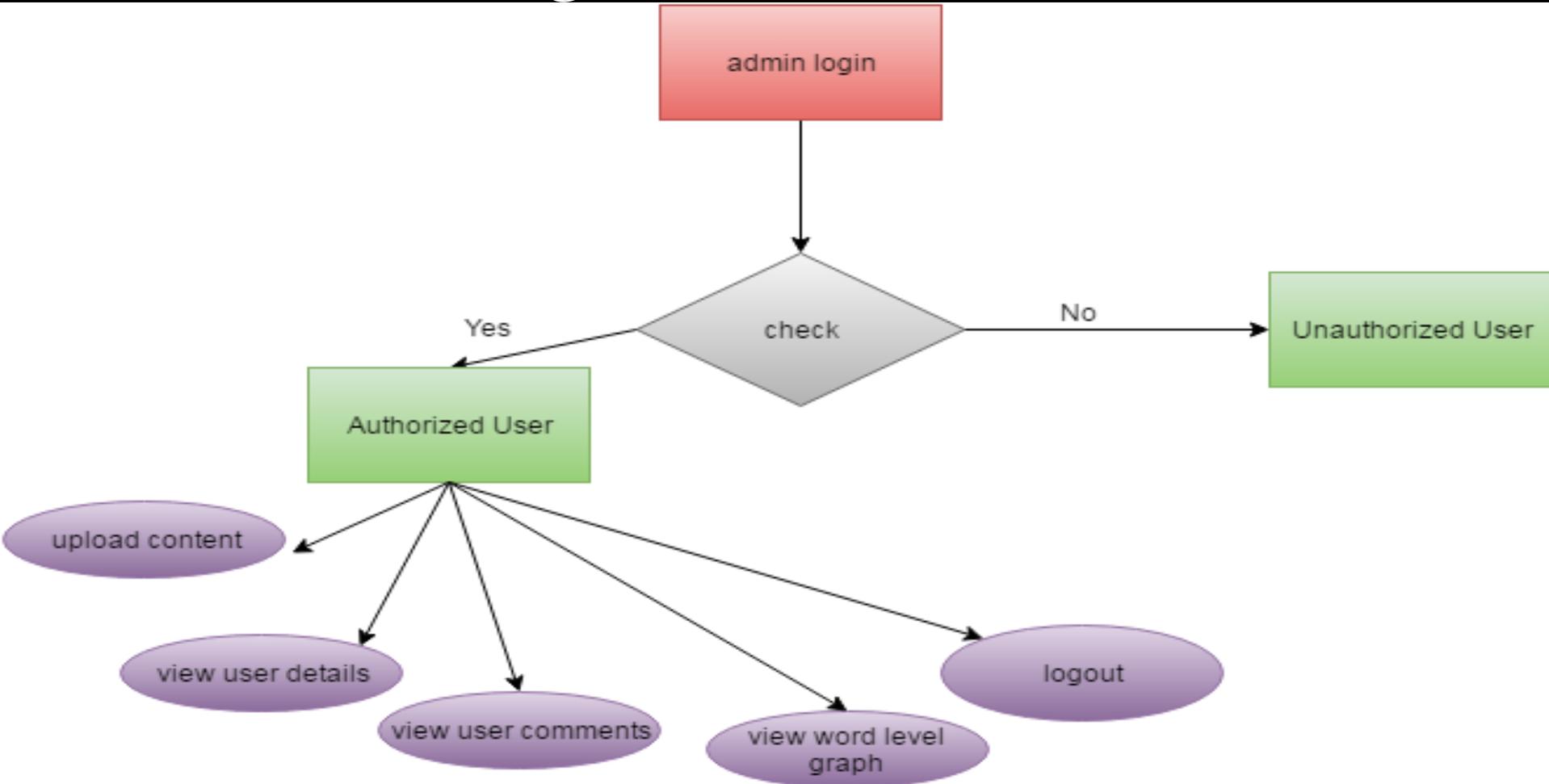
Collaboration diagram



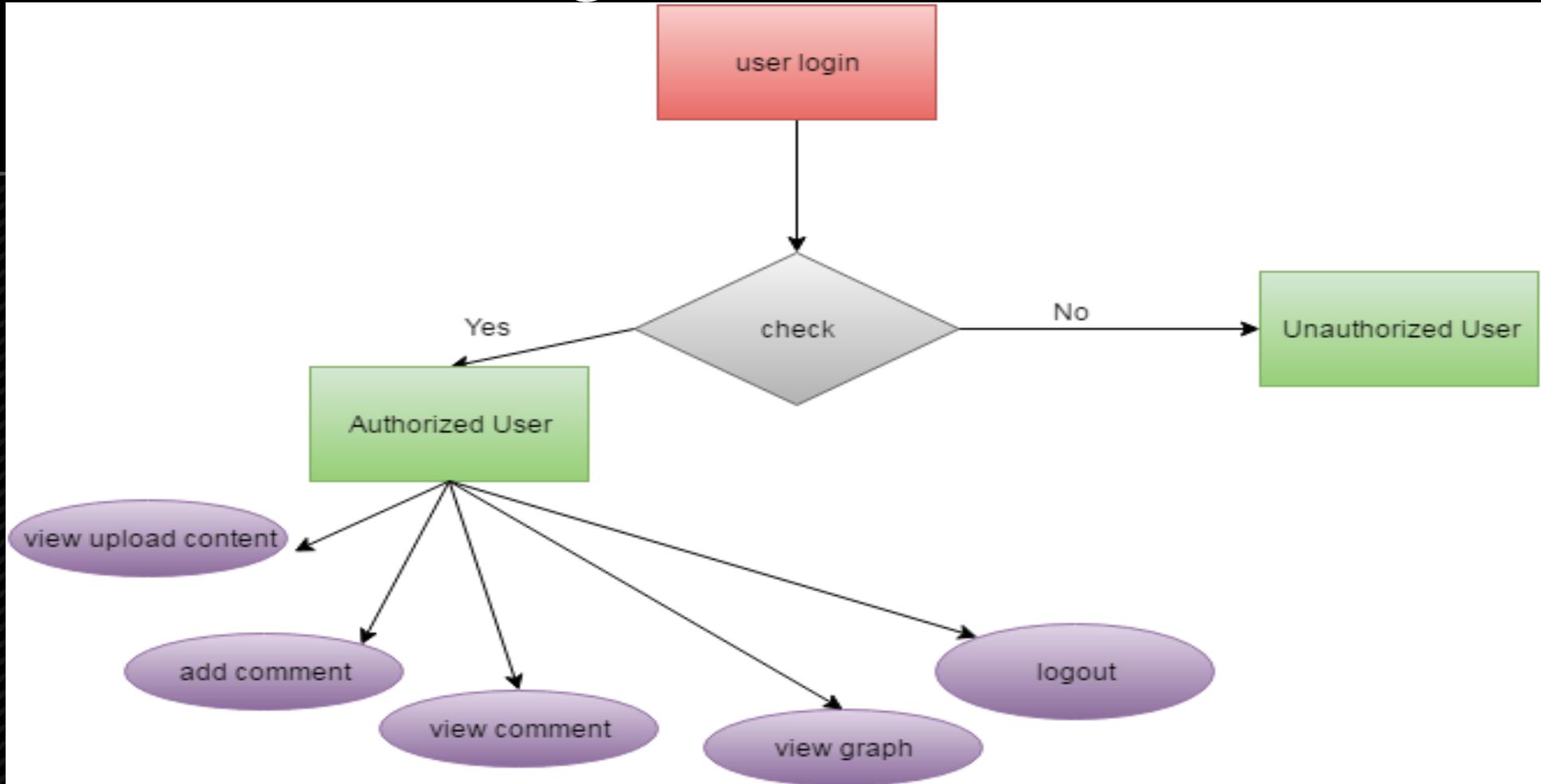
Statechart diagram



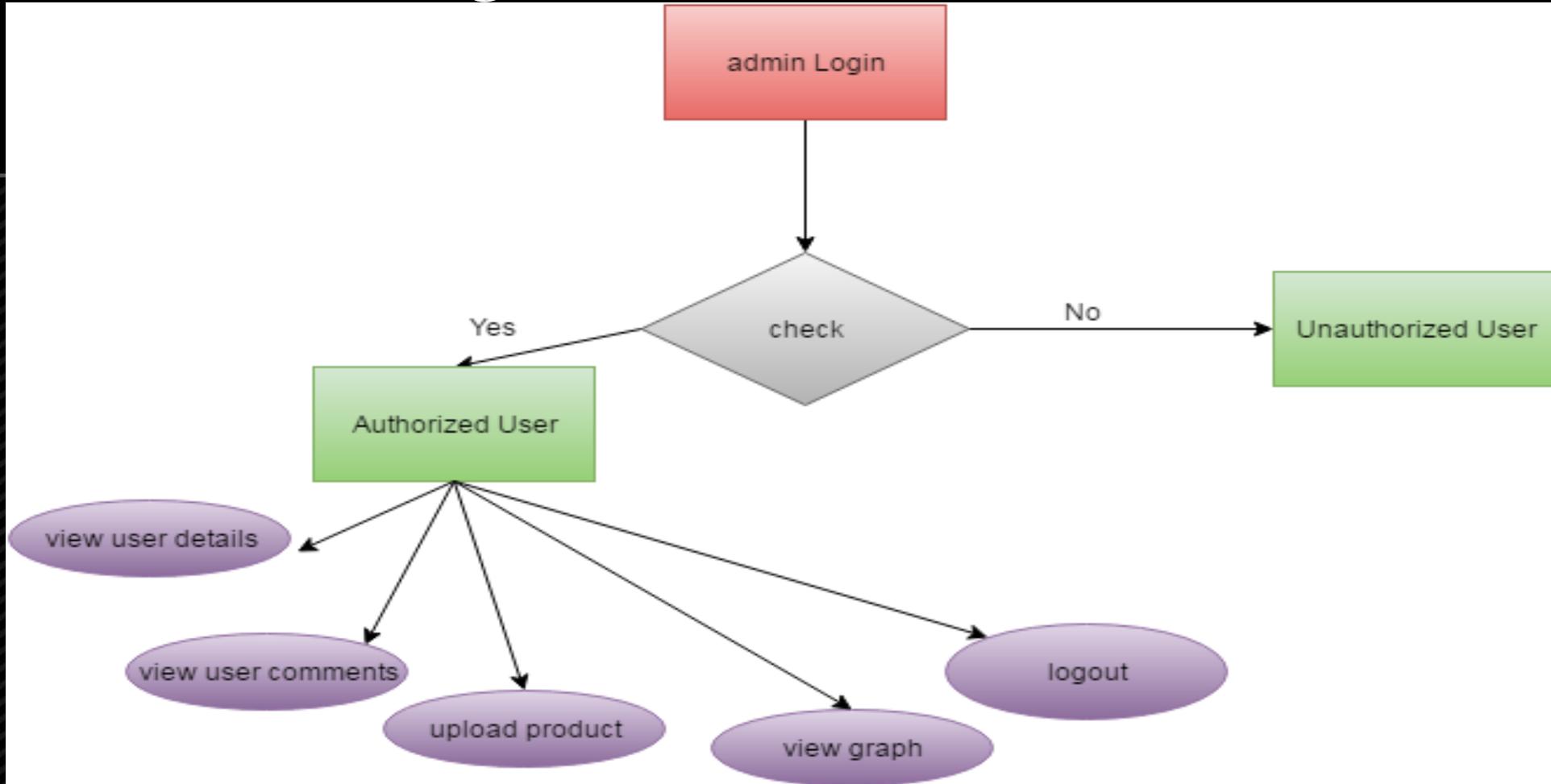
ER-Diagram word level admin



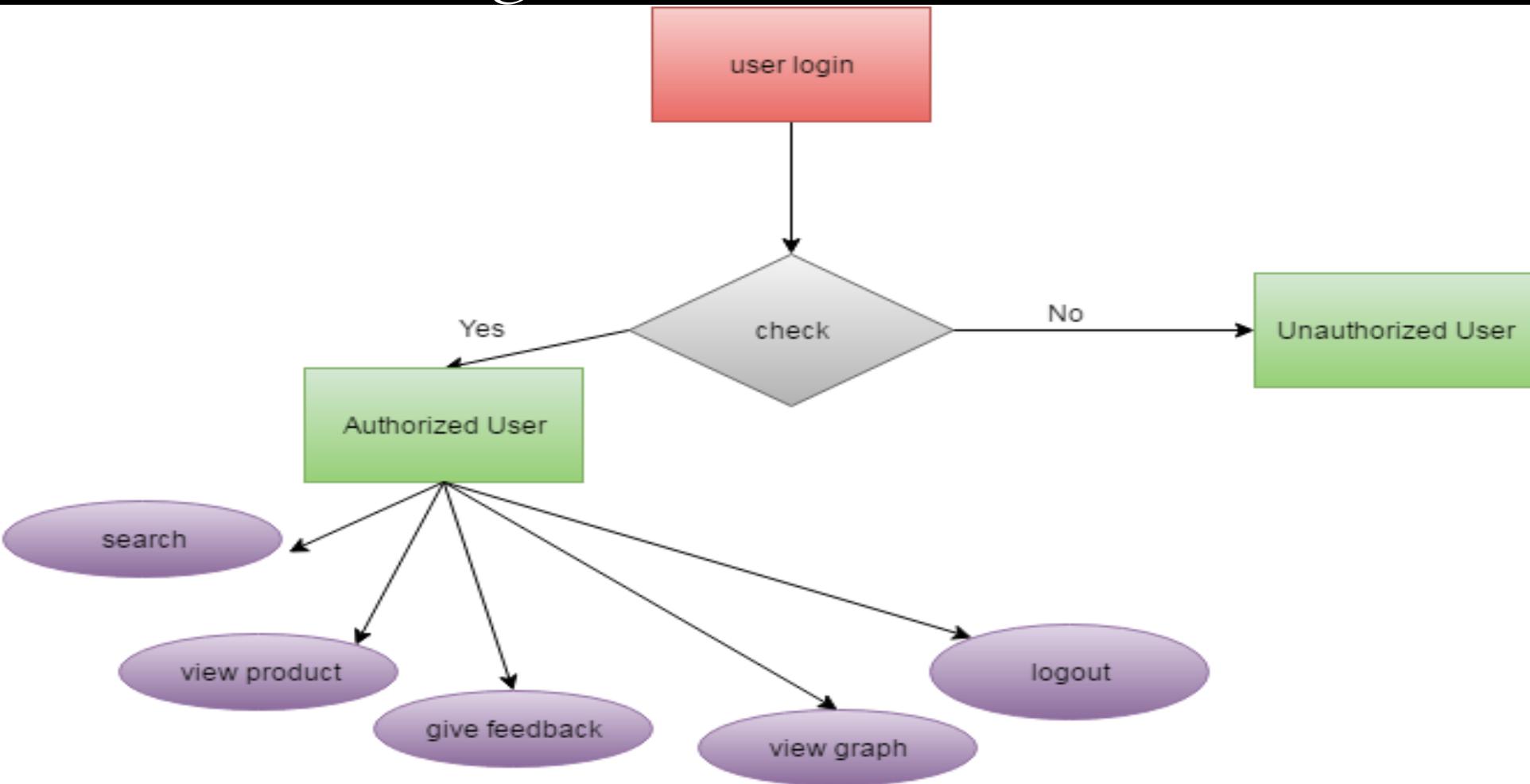
ER-Diagram word level user



ER-Diagram sentence level admin



ER-Diagram sentence level user



Tables

admin

Field	Type	Null	Key	Default
<u>uname</u>	<u>varchar(255)</u>	YES		(NULL)
password	<u>varchar(255)</u>	YES		(NULL)

annotate

Field	Type	Null	Key	Default
<u>labelname</u>	<u>varchar(255)</u>	NO		(NULL)
label	<u>varchar(255)</u>	NO		(NULL)

Tables

comments

Field	Type	Null	Key	Default
username	<u>varchar(255)</u>	YES		(NULL)
id	<u>varchar(22)</u>	YES		(NULL)
name	<u>varchar(255)</u>	YES		(NULL)
category	<u>varchar(255)</u>	YES		(NULL)
profession	<u>varchar(255)</u>	YES		(NULL)
<u>nativeplace</u>	<u>varchar(255)</u>	YES		(NULL)
description	<u>varchar(255)</u>	YES		(NULL)
comment	text	YES		(NULL)
rate	<u>varchar(50)</u>	YES		(NULL)
trust	<u>varchar(255)</u>	YES		(NULL)

details

Field	Type	Null	Key	Default
category	<u>varchar(255)</u>	NO		(NULL)
keyword	<u>varchar(255)</u>	NO		(NULL)
type	<u>varchar(255)</u>	NO		(NULL)
description	<u>varchar(2000)</u>	NO		(NULL)
file	<u>longblob</u>	NO		(NULL)

Tables

opinion

Field	Type	Null	Key	Default
username	<u>varchar</u> (255)	NO		(NULL)
category	<u>varchar</u> (255)	NO		(NULL)
keyword	<u>varchar</u> (255)	NO		(NULL)
type	<u>varchar</u> (255)	NO		(NULL)
comment	<u>varchar</u> (255)	NO		(NULL)
rate	<u>varchar</u> (255)	NO		(NULL)

review

Field	Type	Null	Key	Default
<u>userid</u>	<u>varchar</u> (255)	YES		(NULL)
username	<u>varchar</u> (255)	YES		(NULL)
category	<u>varchar</u> (255)	YES		(NULL)
<u>reviewid</u>	<u>varchar</u> (255)	YES		(NULL)

Tables

sadmin

Field	Type	Null	Key	Default
<u>uname</u>	<u>varchar(255)</u>	YES		(NULL)
password	<u>varchar(255)</u>	YES		(NULL)

slogin

Field	Type	Null	Key	Default
id	<u>int(11)</u>	NO	PRI	(NULL)
<u>user_name</u>	<u>varchar(255)</u>	YES		(NULL)
<u>user_password</u>	<u>varchar(255)</u>	YES		(NULL)
dob	<u>varchar(255)</u>	YES		(NULL)
phone	<u>varchar(255)</u>	YES		(NULL)
address	<u>varchar(255)</u>	YES		(NULL)
email	<u>varchar(255)</u>	YES		(NULL)

Tables

ulogin

Field	Type	Null	Key	Default
id	int(11)	NO	PRI	(NULL)
<u>user_name</u>	<u>varchar(255)</u>	YES		(NULL)
<u>user_password</u>	<u>varchar(255)</u>	YES		(NULL)
dob	<u>varchar(255)</u>	YES		(NULL)
phone	<u>varchar(255)</u>	YES		(NULL)
address	<u>varchar(255)</u>	YES		(NULL)
email	<u>varchar(255)</u>	YES		(NULL)

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
<u>ulogin</u>	0	PRIMARY	1	id	A	3	(NULL)	(NULL)		BTREE

Tables

upload

Field	Type	Null	Key	Default
id	<u>int(11)</u>	YES		(NULL)
name	<u>varchar(255)</u>	YES		(NULL)
category	<u>varchar(255)</u>	YES		(NULL)
profession	<u>varchar(255)</u>	YES		(NULL)
<u>nativeplace</u>	<u>varchar(255)</u>	YES		(NULL)
description	<u>varchar(255)</u>	YES		(NULL)
image	<u>longblob</u>	YES		(NULL)

CONCLUSION



- We learn sentiment-specific word embeddings named as sentiment embeddings in this paper. Different from majority of exiting studies that only encode word contexts in word embeddings, we factor in sentiment of texts to facilitate the ability of word embeddings in capturing word similarities in terms of sentiment semantics.
- As a result, the words with similar contexts but opposite sentiment polarity labels like “good” and “bad” can be separated in the sentiment embedding space. We introduce several neural networks to effectively encode context and sentiment level informations simultaneously into word embeddings in a unified way.

**ANY
QUESTIONS?**

Thank You!

