FYP PRESENTATION

PROJECT SCE16-0152
“WEB-BASED MATHEMATICAL DOCUMENT RETRIEVAL FOR MOBILE ANDROID APPLICATION”

BY: DEKA AULIYA AKBAR - U1323056K

Supervisor: A/P Hui Siu Cheung

Examiner: Prof. Lam Kwok Yan

Link to Presentation Slides: http://bit.ly/fyp_deka

GUIDE TO THE PRESENTATION

- Section 1: Introduction
- Section 2: System Design
- Section 3: Server-side Development
- Section 4: Mathematical Document Retrieval
- Section 5: Android Development
- Section 6: Optical Character Recognition
- Section 7: Demonstration
- Section 8: Conclusion and Future Recommendations

1.

INTRODUCTION

Background and Motivation, Project Objectives, Approaches, Related Knowledge

“Mobile phones are misnamed. They should be called gateways to human knowledge.”

- Ray Kurzweil

BACKGROUND AND MOTIVATIONS

- **Why mathematics?**
  - Mathematics is an important subject to be learned by everyone
  - Mathematical content is **non-trivial**: complex structure
- **Why mobile phones?**
  - Smartphones are prevalent in human lives
    - convenience, portability, connectivity, powerful
  - promotes mobile and interactive learning
- **Why Android?**
  - Holds 81% market share of the world population (Gartner Q4 2016)
  - Experimented tools are only available for Android devices

BACKGROUND AND MOTIVATIONS

- Relevant work: previous students' work on mathematical document retrieval
- Problems:
  - different system design and architecture to the new MathQA system
  - new MathQA system does not have **mathematical document retrieval**
  - no clear **evaluation** on the formula-based retrieval performance
  - **no mobile application**
  - LaTeX content contains **erroneous** and **unstandardised**

PROJECT OBJECTIVES

- Develop a **Web Server**, which provides:
  - educational **mathematics contents**
    - mathematical content database
    - fix erroneous and unstandardised LaTeX content
  - **search services** for retrieving mathematical documents
    - text-based and formula-based retrieval
- Develop an **Android Application**, which provides:
  - **user interface** (UI) for **accessing** and **searching** mathematical contents
    - display LaTeX
    - access MathQA database content
  - incorporate **camera** and **OCR** services to perform search through **document image recognition**

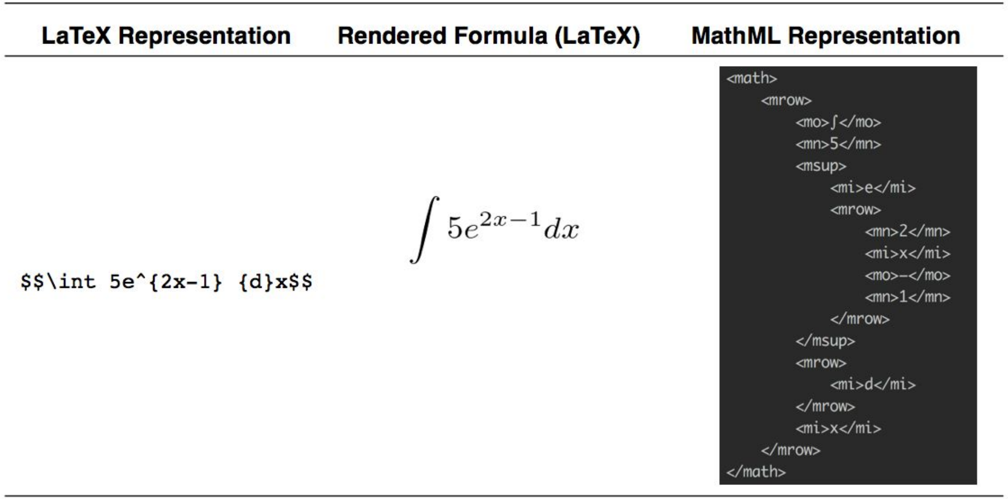
MathQA

PROJECT DEVELOPMENT APPROACHES

1. **Exploration** Phase:
   a. Mathematical Document Retrieval techniques in [1] and [2]
   b. Open-source technologies for developing web and mobile applications and OCR in mobile phones
2. **Development** Phase:
   a. web services: database and mathematical document retrieval
   b. Android application: access the mathematics educational content

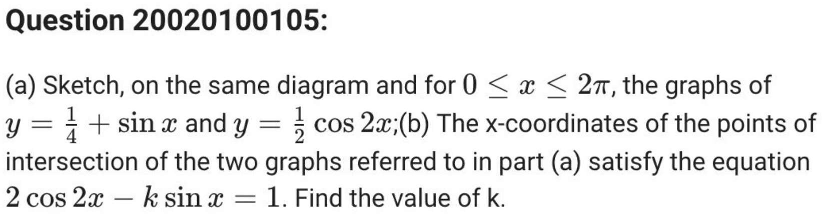
RELATED KNOWLEDGE - MATHEMATICAL DOCUMENTS

- documents that contain both formula and textual content
  - textual representation: alphanumeric characters
  - formula representation: LaTeX and MathML
- in this project, mathematical documents refer to mathematical questions



LaTeX Representation	Rendered Formula (LaTeX)	MathML Representation

`$$\int 5e^{2x-1} {d}x$$`	$\int 5e^{2x-1}dx$	<math>
		  <mrow>
		    <mo>∫</mo>
		    <mn>5</mn>
		    <msup>
		      <mi>e</mi>
		      <mrow>
		        <mn>2</mn>
		        <mi>x</mi>
		        <mo>−</mo>
		        <mn>1</mn>
		      </mrow>
		    </msup>
		    <mrow>
		      <mi>d</mi>
		    </mrow>
		    <mi>x</mi>
		  </mrow>
		</math>



Question 20020100105:

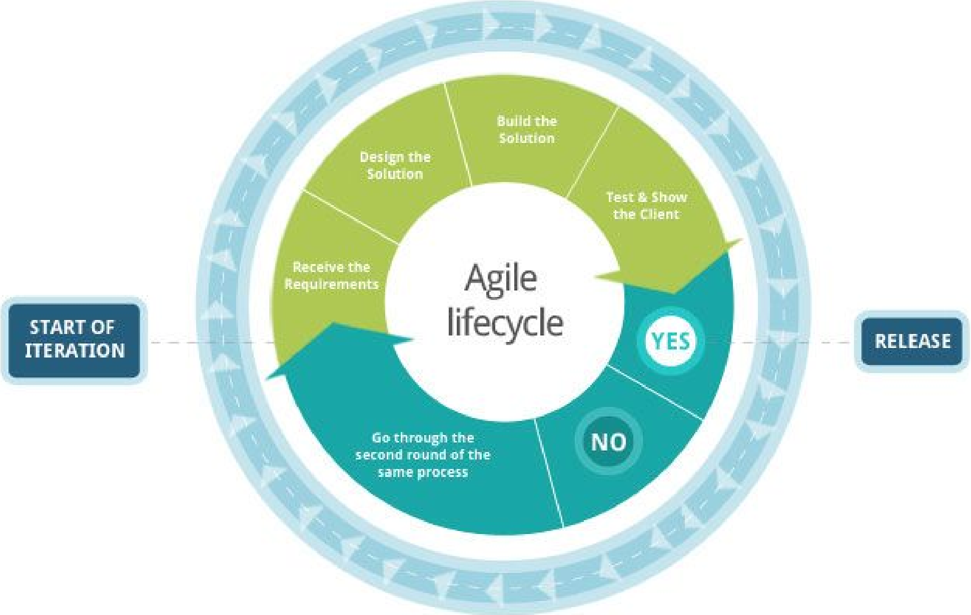
(a) Sketch, on the same diagram and for $0 \leq x \leq 2\pi$, the graphs of $y = \frac{1}{4} + \sin x$ and $y = \frac{1}{2}\cos 2x$;(b) The x-coordinates of the points of intersection of the two graphs referred to in part (a) satisfy the equation $2\cos 2x - k\sin x = 1$. Find the value of k.

RELATED KNOWLEDGE - OPTICAL CHARACTER RECOGNITION (OCR)

- OCR systems enable **automatic pattern recognition** of **alphanumeric** and **handwritten characters** in document images
- Image Pre-processing:
  - Enhancement, Grayscaling, Image Segmentation (Binarisation), De-skewing
- Investigated Open-Source OCR Technologies
  - Image Pre-processing: Leptonica and Catalano
  - OCR Engines: Tesseract, Google Text API

RELATED KNOWLEDGE - SOFTWARE ENGINEERING

- Agile Software Development Lifecycle



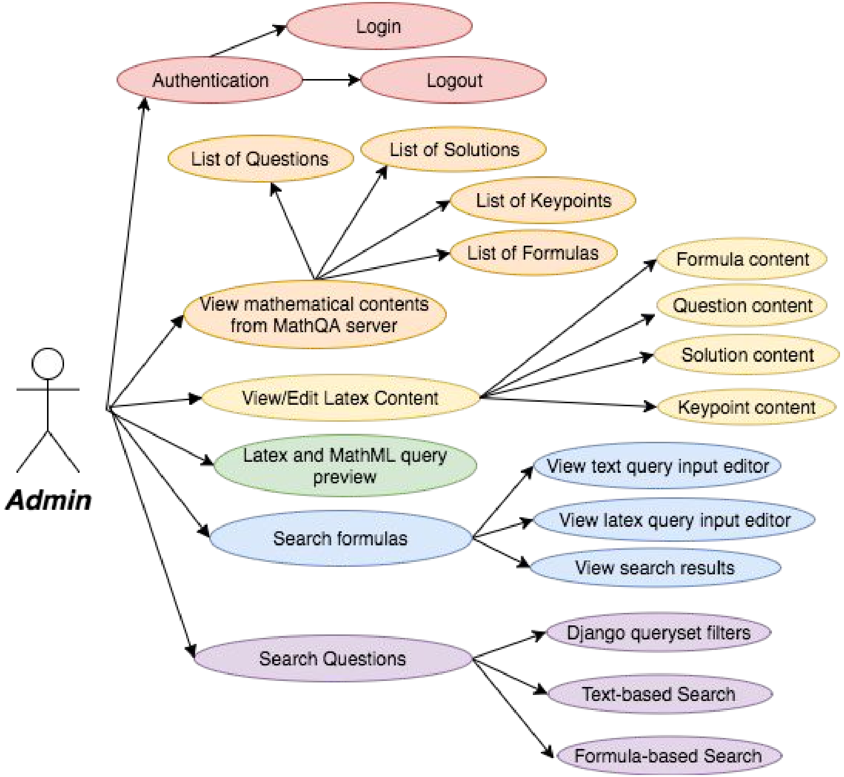
- SOLID and Don’t Repeat Yourself (DRY) Principles

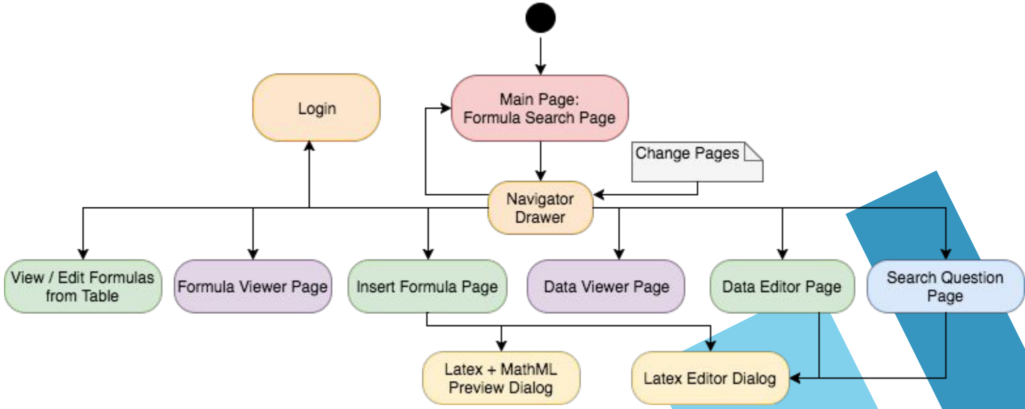
2.

SYSTEM DESIGN

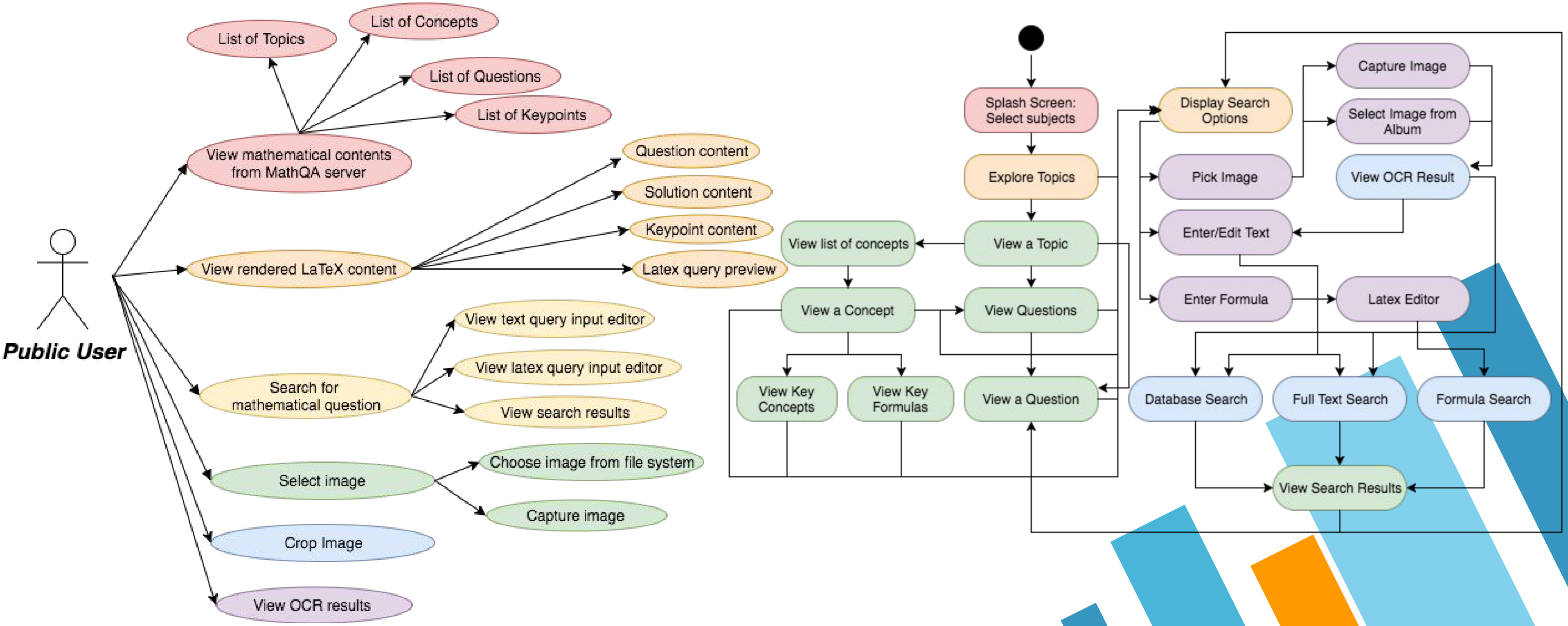
Use Cases, Architecture, Database Design

SERVER (ADMIN) USE CASES: WEB SERVER

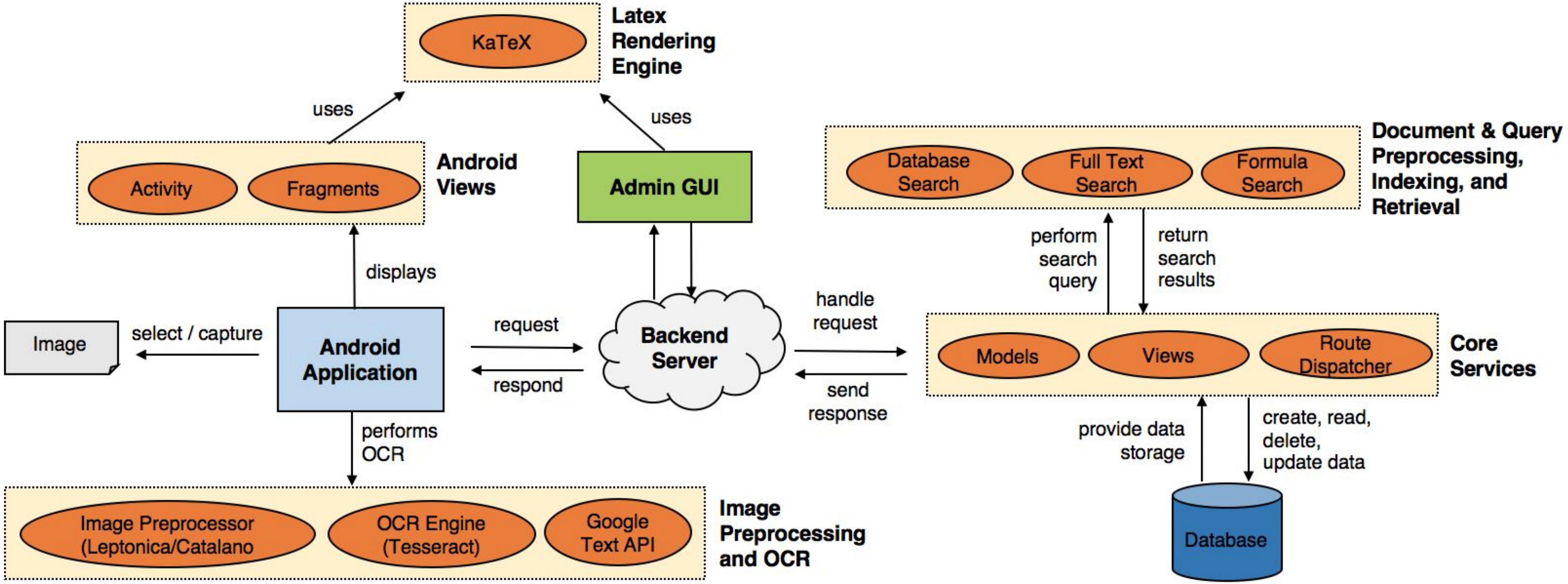




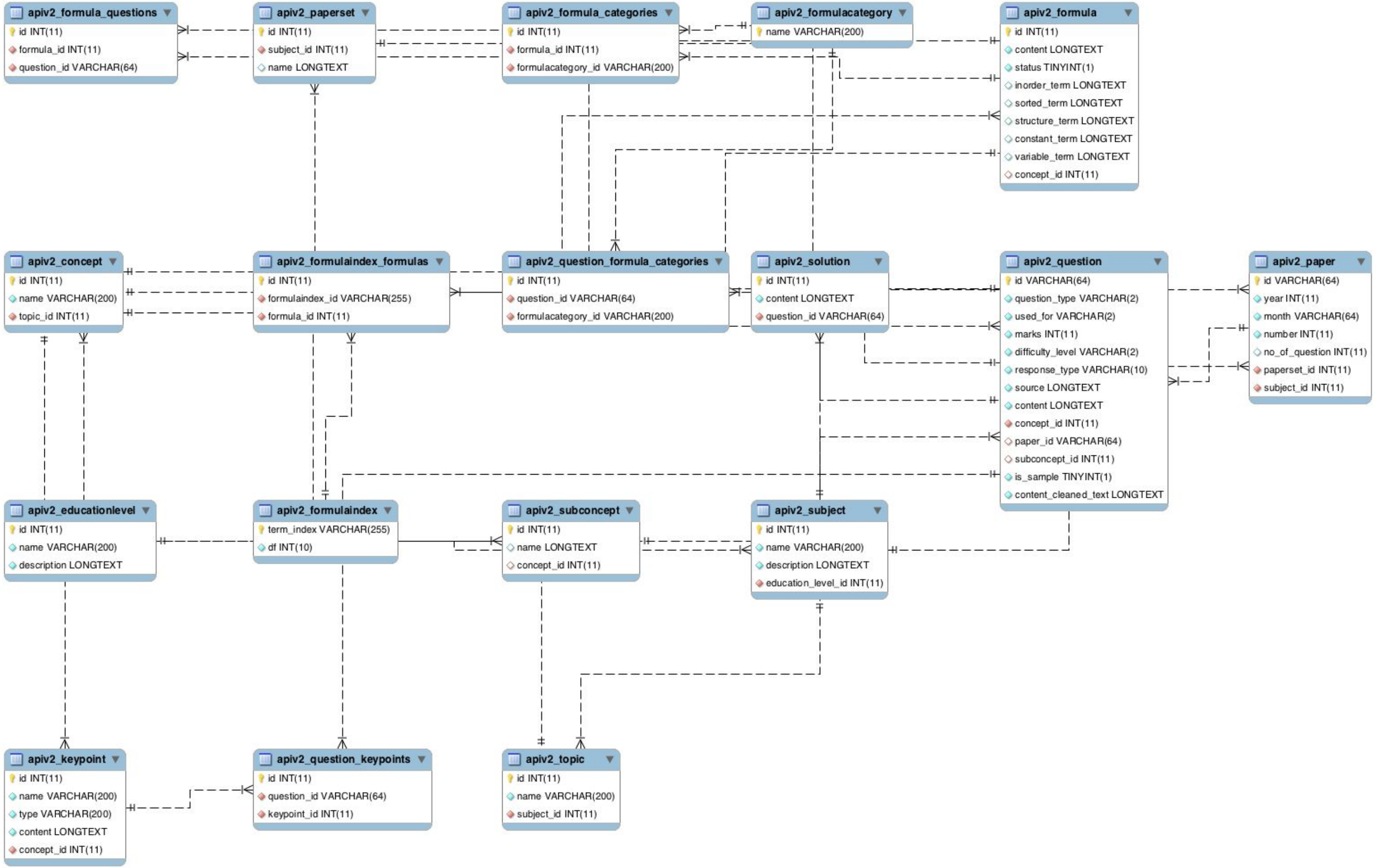
PUBLIC USER USE CASES: ANDROID



OVERALL ARCHITECTURE



DATABASE DESIGN



apiv2_formula_questions
- id INT(11)
- formula_id INT(11)
- question_id VARCHAR(64)

apiv2_paperset
- id INT(11)
- subject_id INT(11)
- name LONGTEXT

apiv2_formula_categories
- id INT(11)
- formula_id INT(11)
- formulacategory_id VARCHAR(200)

apiv2_formulacategory
- name VARCHAR(200)

apiv2_formula
- id INT(11)
- content LONGTEXT
- status TINYINT(1)
- inorder_term LONGTEXT
- sorted_term LONGTEXT
- structure_term LONGTEXT
- constant_term LONGTEXT
- variable_term LONGTEXT
- concept_id INT(11)

apiv2_concept
- id INT(11)
- name VARCHAR(200)
- topic_id INT(11)

apiv2_formulaindex_formulas
- id INT(11)
- formulaindex_id VARCHAR(255)
- formula_id INT(11)

apiv2_question_formula_categories
- id INT(11)
- question_id VARCHAR(64)
- formulacategory_id VARCHAR(200)

apiv2_solution
- id INT(11)
- content LONGTEXT
- question_id VARCHAR(64)

apiv2_question
- id VARCHAR(64)
- question_type VARCHAR(2)
- used_for VARCHAR(2)
- marks INT(11)
- difficulty_level VARCHAR(2)
- response_type VARCHAR(10)
- source LONGTEXT
- content LONGTEXT
- concept_id INT(11)
- paper_id VARCHAR(64)
- subconcept_id INT(11)
- is_sample TINYINT(1)
- content_cleaned_text LONGTEXT

apiv2_paper
- id VARCHAR(64)
- year INT(11)
- month VARCHAR(64)
- number INT(11)
- no_of_question INT(11)
- paperset_id INT(11)
- subject_id INT(11)

apiv2_educationlevel
- id INT(11)
- name VARCHAR(200)
- description LONGTEXT

apiv2_formulaindex
- term_index VARCHAR(255)
- df INT(10)

apiv2_subconcept
- id INT(11)
- name LONGTEXT
- concept_id INT(11)

apiv2_subject
- id INT(11)
- name VARCHAR(200)
- description LONGTEXT
- education_level_id INT(11)

apiv2_keypoint
- id INT(11)
- name VARCHAR(200)
- type VARCHAR(200)
- content LONGTEXT
- concept_id INT(11)

apiv2_question_keypoints
- id INT(11)
- question_id VARCHAR(64)
- keypoint_id INT(11)

apiv2_topic
- id INT(11)
- name VARCHAR(200)
- subject_id INT(11)

3.

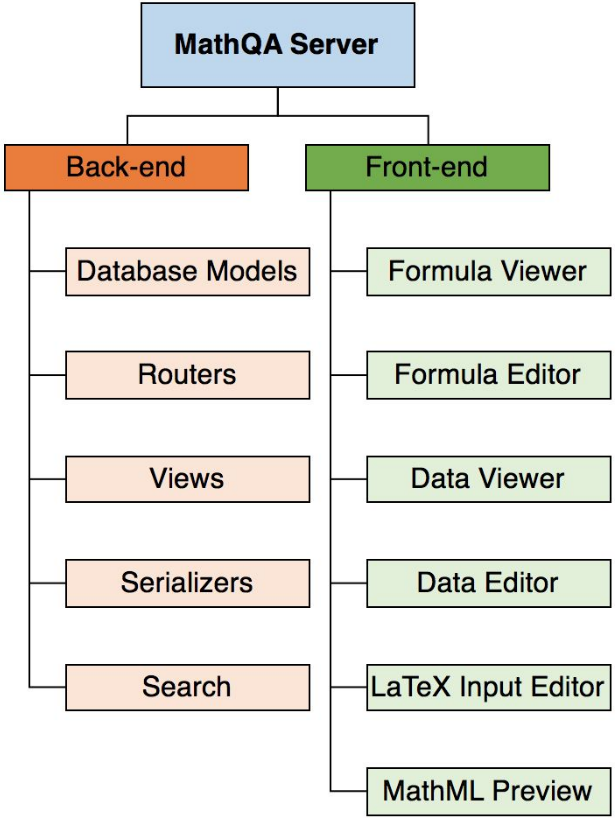
SERVER-SIDE DEVELOPMENT

Tools, Web-Server Architecture, Backend, Frontend

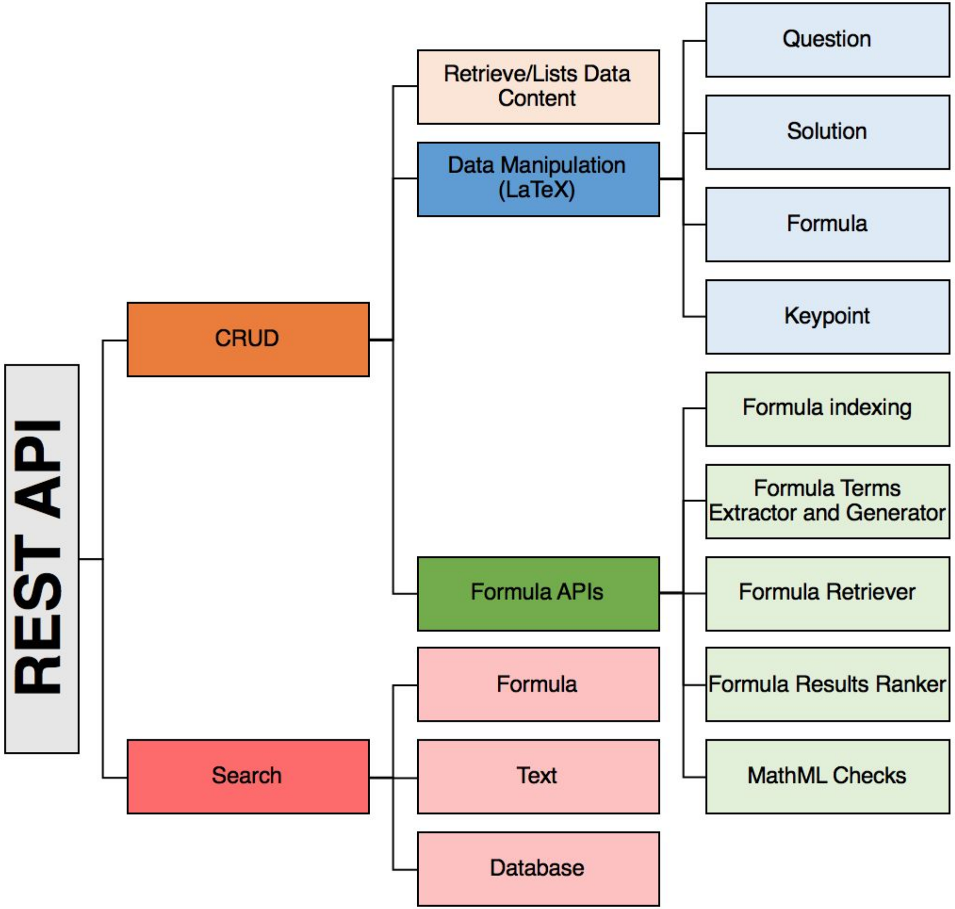
WEB-SERVER DEVELOPMENT TOOLS

- Language: Python 2.7
- Tools and Libraries
  - Back-end:
    - Django 1.10
    - MySQL
    - Django REST framework
    - latex2mathml parser
    - Django Haystack
    - Whoosh
    - Natural Language Toolkit (NLTK)
  - Front-end:
    - HTML, CSS and AngularJS Material
    - KaTeX

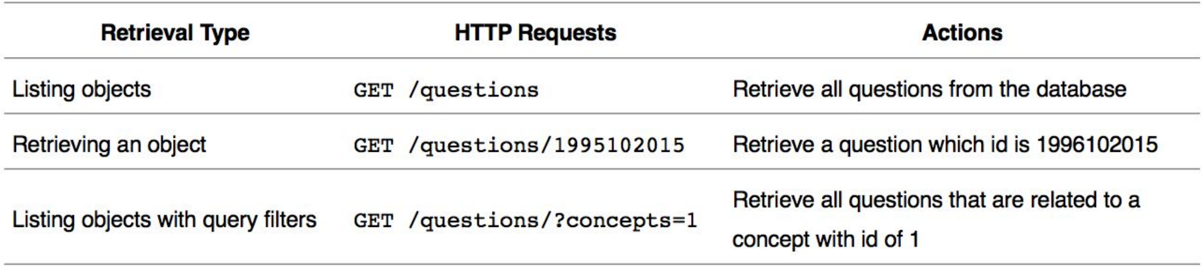
WEB-SERVER ARCHITECTURE



WEB-SERVER DEVELOPMENT - BACKEND

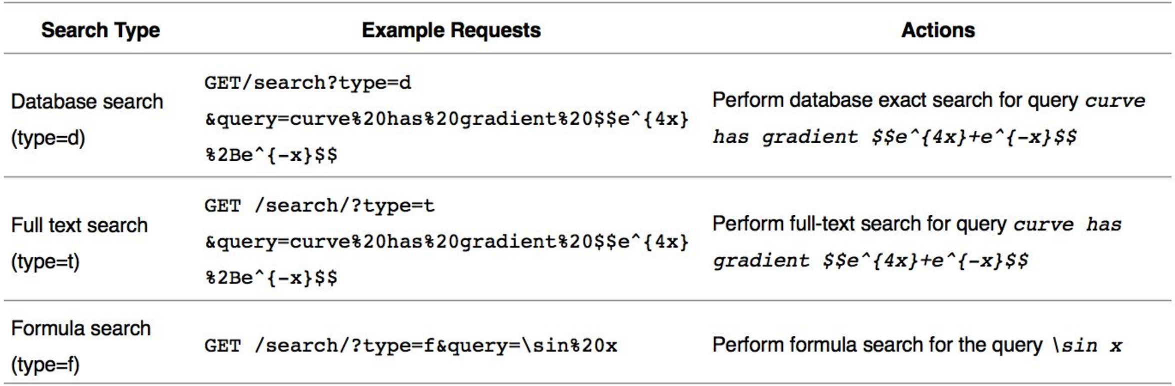


WEB-SERVER DEVELOPMENT - BACKEND



Retrieval Type	HTTP Requests	Actions
Listing objects	`GET /questions`	Retrieve all questions from the database
Retrieving an object	`GET /questions/1995102015`	Retrieve a question which id is 1996102015
Listing objects with query filters	`GET /questions/?concepts=1`	Retrieve all questions that are related to a concept with id of 1

Read-Only Access APIs Examples



Search Type	Example Requests	Actions
Database search (type=d)	`GET/search?type=d &query=curve%20has%20gradient%20$$e^{4x} %2Be^{-x}$$`	Perform database exact search for query `curve has gradient $$e^{4x}+e^{-x}$$`
Full text search (type=t)	`GET /search/?type=t &query=curve%20has%20gradient%20$$e^{4x} %2Be^{-x}$$`	Perform full-text search for query `curve has gradient $$e^{4x}+e^{-x}$$`
Formula search (type=f)	`GET /search/?type=f&query=\sin%20x`	Perform formula search for the query `\sin x`

Search APIs

WEB-SERVER DEVELOPMENT - FRONTEND

- Admin Graphical User Interface (GUI) is developed for admins to:
  - **create / update /delete** LaTeX formulas
  - **inspect** and **manipulate** LaTeX contents
  - **evaluate** retrieval performance
- Components include:
  - **Data services:** retrieves data from server database
  - **View controllers:** controls UI behaviour and logic between components
  - **HTML templates**: provide the UI components of the GUI

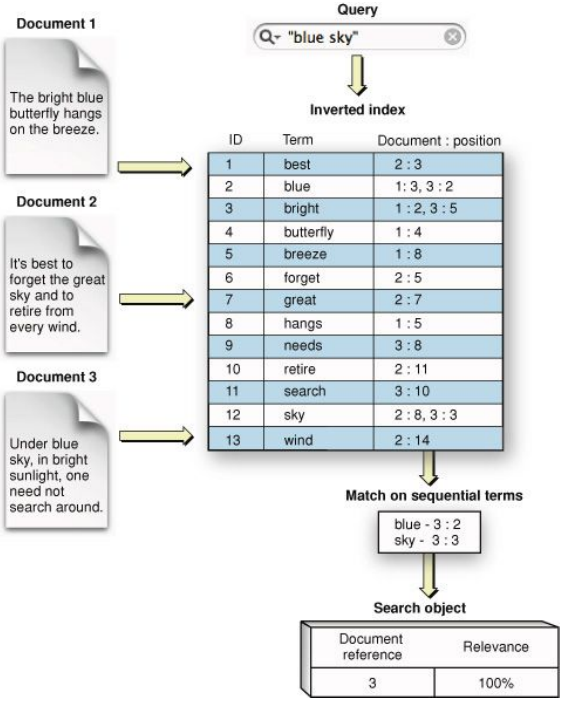
4.

MATHEMATICAL DOCUMENT RETRIEVAL

Definition, Text-based Retrieval, Formula-based Retrieval, Formula-based Retrieval Evaluation

MATHEMATICAL DOCUMENT RETRIEVAL

» Separate the retrieval into two phases: Text-based Retrieval and Formula-based Retrieval



Document 1

The bright blue butterfly hangs on the breeze.

Document 2

It's best to forget the great sky and to retire from every wind.

Document 3

Under blue sky, in bright sunlight, one need not search around.

Query

"blue sky"

Inverted index

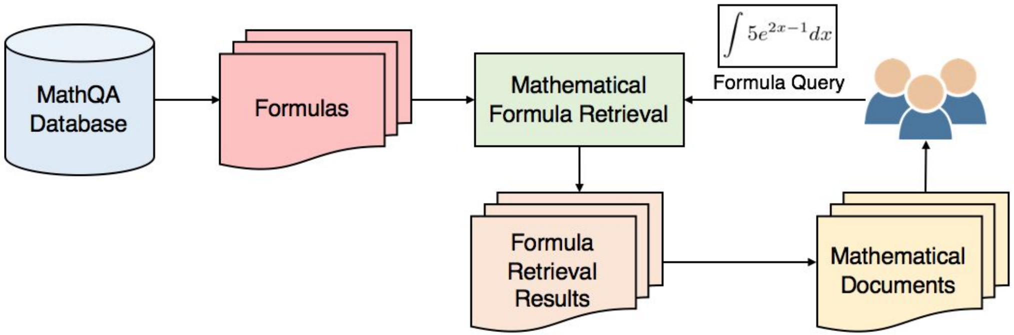
ID	Term	Document : position
1	best	2 : 3
2	blue	1: 3, 3 : 2
3	bright	1 : 2, 3 : 5
4	butterfly	1 : 4
5	breeze	1 : 8
6	forget	2 : 5
7	great	2 : 7
8	hangs	1 : 5
9	needs	3 : 8
10	retire	2 : 11
11	search	3 : 10
12	sky	2 : 8, 3 : 3
13	wind	2 : 14

Match on sequential terms

blue - 3 : 2
sky - 3 : 3

Search object

Document reference	Relevance
3	100%



TEXT BASED RETRIEVAL

Database Search

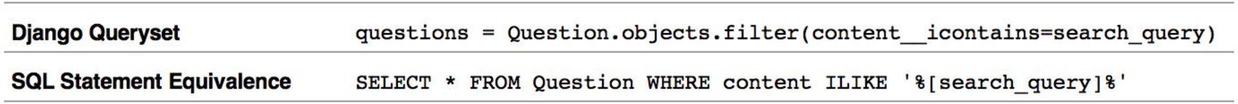
- implemented using Django **queryset** icontains **filter**
- expected result: **exact** (very similar) match with the database
  - order of term appearance matters
- fewer results (strict)
- optimised for:
  - looking exact match query results
  - can supply exact LaTeX syntax directly as the query

Full-text Search

- implemented using **NLTK** and **Haystack**
- requires **document pre-processing**, **term-indexing** and **querying**
- expected result: get more **relevant** results rather than exact matching
  - order of term appearance does not matter
- optimised for:
  - finding most relevant results to the query
  - allowing term suffixes and terms appearing in different order in the query

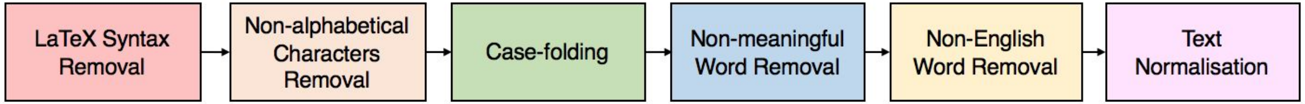
TEXT BASED RETRIEVAL

Exact Database Search

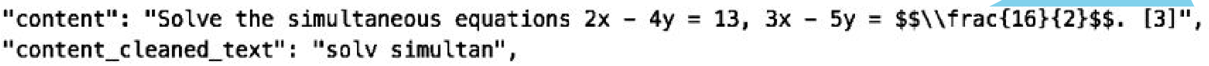


Django Queryset	`questions = Question.objects.filter(content__icontains=search_query)`
SQL Statement Equivalence	`SELECT * FROM Question WHERE content ILIKE '%[search_query]%'`

Document Pre-processing
(Full-Text Search)

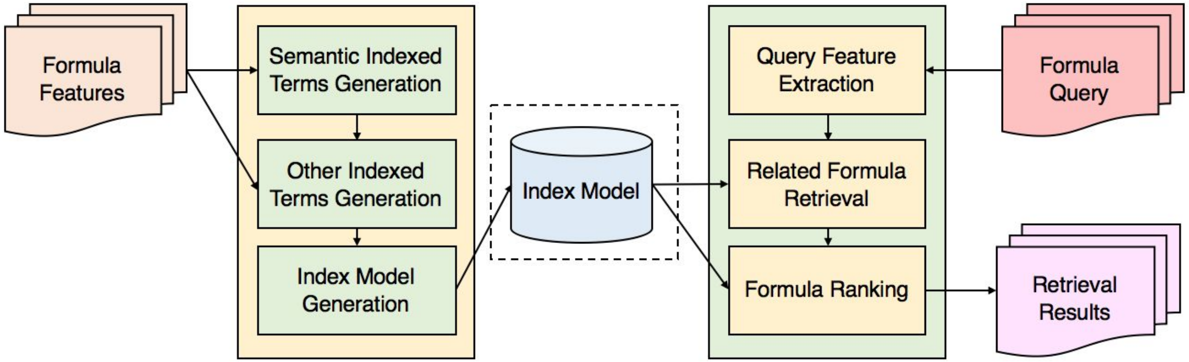


Pre-processed Text
(Full-Text Search)



```
"content": "Solve the simultaneous equations 2x – 4y = 13, 3x – 5y = $$\\frac{16}{2}$$. [3]",
"content_cleaned_text": "solv simultan",
```

FORMULA BASED RETRIEVAL



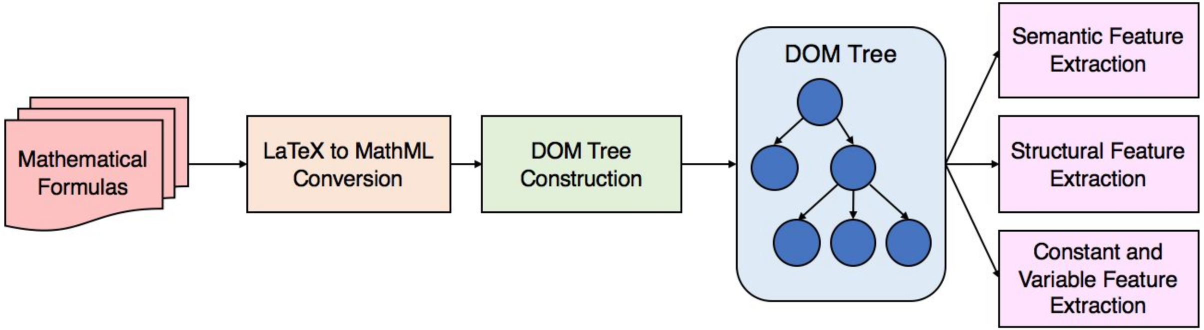
Formula-based Retrieval Steps

1. Formula Features Extraction
2. Formula Indexing
3. Formula Retrieval
4. Formula Ranking
5. Mathematical Document Retrieval

FORMULA BASED RETRIEVAL

1. **Formula Feature Extraction**

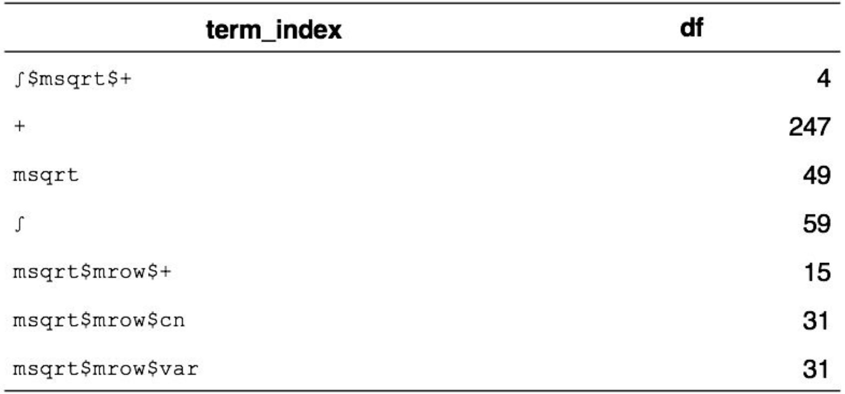
- Formula Features:
    - Semantic Features: capture semantic information → mathematical functions and operators
        - in-order: 2, 3, 4-grams of the original ordered sequence
        - sorted: lexicographically sorted semantic feature
    - Structural Features
    - Constant Features
    - Variable Features


Mathematical Formulas
LaTeX to MathML Conversion
DOM Tree Construction
DOM Tree
Semantic Feature Extraction
Structural Feature Extraction
Constant and Variable Feature Extraction

FORMULA BASED RETRIEVAL

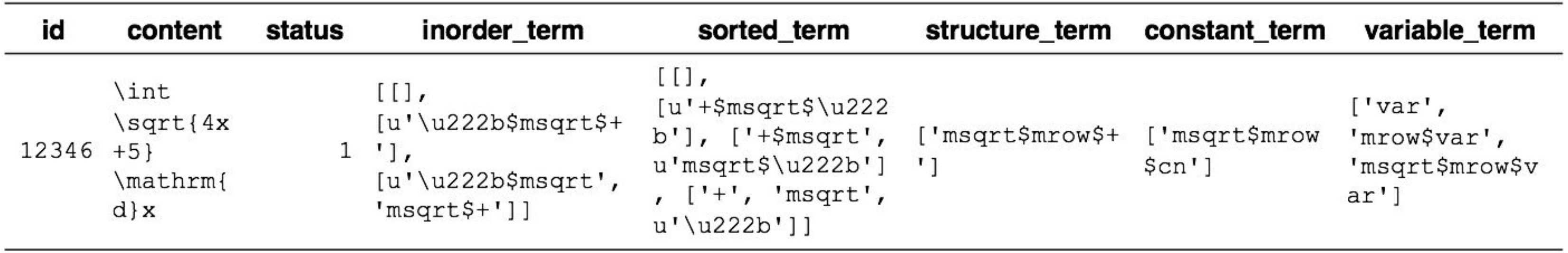
2. Formula Indexing

- indexes formula terms
  - maps formula term to list of formulas containing the term
- create / update Formula and FormulaIndex tables



term_index	df
∫$msqrt$+	4
+	247
msqrt	49
∫	59
msqrt$mrow$+	15
msqrt$mrow$cn	31
msqrt$mrow$var	31

Formula Index Table



id	content	status	inorder_term	sorted_term	structure_term	constant_term	variable_term
12346	\int \sqrt{4x +5} \mathrm{d}x	1	[[], [u'\u222b$msqrt$+'], [u'\u222b$msqrt', 'msqrt$+']]	[[], [u'+$msqrt$\u222b'], ['+$msqrt', u'msqrt$\u222b'], ['+', 'msqrt', u'\u222b']]	['msqrt$mrow$+']	['msqrt$mrow$cn']	['var', 'mrow$var', 'msqrt$mrow$var']

Formula Table

FORMULA BASED RETRIEVAL

3. **Formula Retrieval**

- uses top-k retrieval technique
- involves query processing, related formula retrieval and ranking
- set union operation between formula and query terms

4. **Formula Ranking**

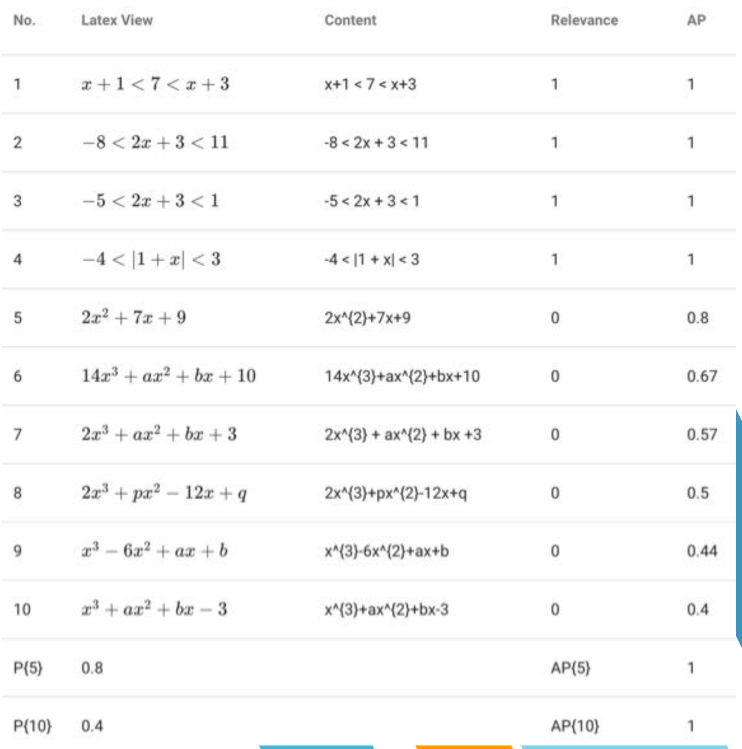
- computes the similarity score between related formulas
  - uses set intersection and set difference operations between formula and query
  - assign more weights to semantic terms
- related formulas is sorted in decreasing order of the similarity score

5. **Mathematical Document Retrieval**

- many-to-many relationship is established between question and formula
- mathematical questions that contain the ranked relevant formulas are retrieved and returned

FORMULA-BASED RETRIEVAL EVALUATION

- **Test Data Preparation**:
  - ◊ inserted 508 formulas for some formula categories
  - ◊ created 52 formula test queries
- **Tools**: Admin GUI
- **Methodology**:
  1. Conduct formula search using the formula test queries
  2. Extracted top-10 formula results and assign relevance label (0 or 1)
  3. Compute Precision@K and Mean Average Precision@K with K=5 and 10



No.	Latex View	Content	Relevance	AP
1	$x+1<7<x+3$	x+1 < 7 < x+3	1	1
2	$-8<2x+3<11$	-8 < 2x + 3 < 11	1	1
3	$-5<2x+3<1$	-5 < 2x + 3 < 1	1	1
4	$-4<\|1+x\|<3$	-4 < |1 + x| < 3	1	1
5	$2x^2+7x+9$	2x^{2}+7x+9	0	0.8
6	$14x^3+ax^2+bx+10$	14x^{3}+ax^{2}+bx+10	0	0.67
7	$2x^3+ax^2+bx+3$	2x^{3} + ax^{2} + bx +3	0	0.57
8	$2x^3+px^2-12x+q$	2x^{3}+px^{2}-12x+q	0	0.5
9	x^3-6x^2+ax+b	x^{3}-6x^{2}+ax+b	0	0.44
10	x^3+ax^2+bx-3	x^{3}+ax^{2}+bx-3	0	0.4
P(5)	0.8		AP(5)	1
P(10)	0.4		AP(10)	1

$$MAP = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{m_i}\sum_{k=1}^{m_i} Precision(rank_{ik})$$

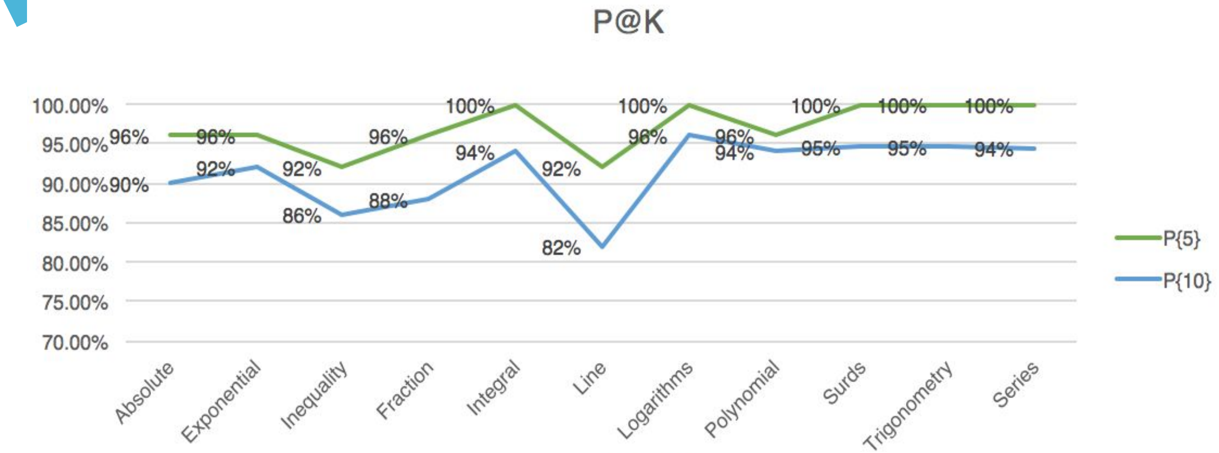
$$Precision = \frac{\#\text{relevant items}}{\#\text{retrieved items}}$$

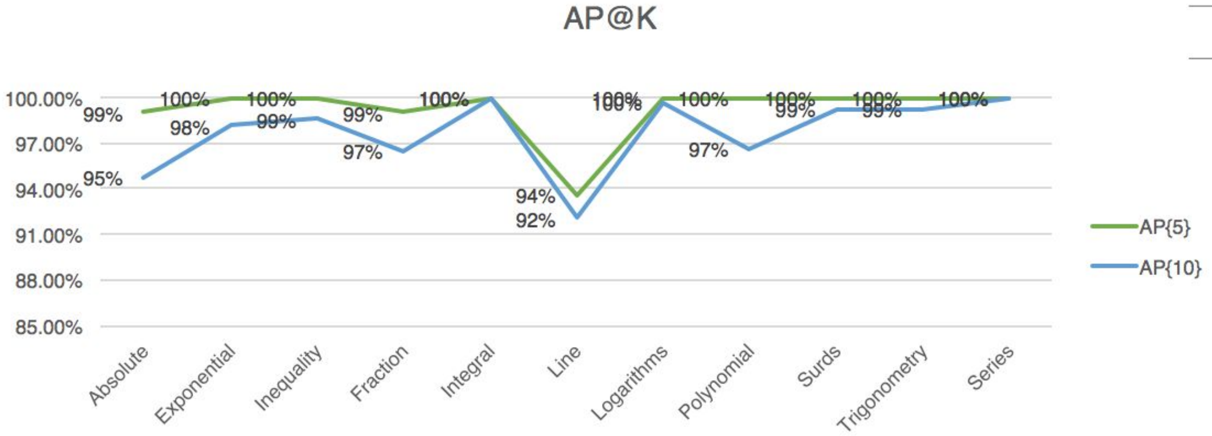
Q: Test queries

m_i: number of items retrieved for query i

Precision($rank_{ik}$): ratio of the top k retrieved items that are relevant

FORMULA-BASED RETRIEVAL PERFORMANCE



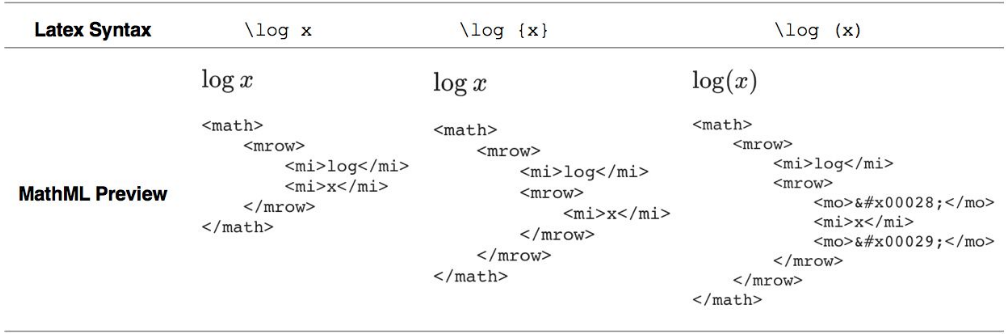


Overall Result

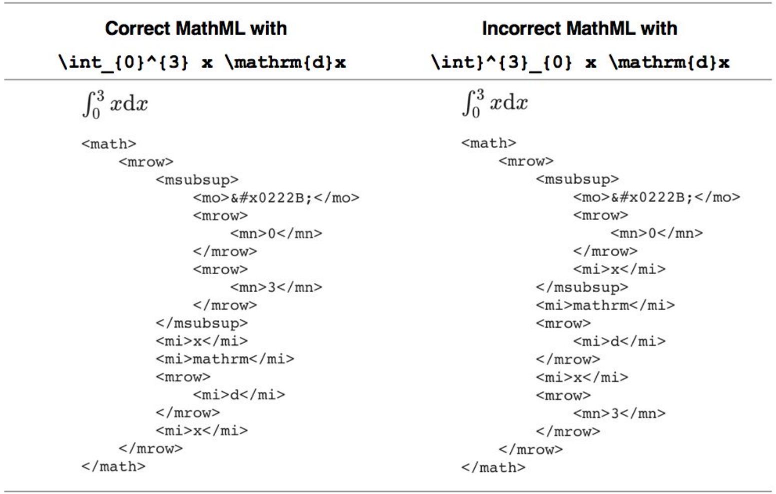
AP {5}	AP {10}	MAP {5}	MAP {10}
0.965	0.9139	0.9919	0.9749777724

FORMULA-BASED RETRIEVAL DISCUSSIONS

- **Problems**:
  - initially, poor formula-based retrieval accuracy due to:
    - LaTeX **syntax errors** and **incompatible** LaTeX syntax e.g. ‘<’ operator
    - **Unparsable** LaTeX to MathML
    - **Non-standardised** LaTeX formulas: similar meaning but different LaTeX representation
- **Solution**:
  - encode into HTML entities
  - use admin GUI’s data editor to fix LaTeX errors
  - use rules defined in the report when fixing / entering LaTeX



Latex Syntax	`\log x`	`\log {x}`	`\log (x)`
MathML Preview	$\log x$ <math> <mrow> <mi>log</mi> <mi>x</mi> </mrow> </math>	$\log x$ <math> <mrow> <mi>log</mi> <mrow> <mi>x</mi> </mrow> </mrow> </math>	$\log(x)$ <math> <mrow> <mi>log</mi> <mrow> <mo>&#x00028;</mo> <mi>x</mi> <mo>&#x00029;</mo> </mrow> </mrow> </math>



Correct MathML with `\int_{0}^{3} x \mathrm{d}x`	**Incorrect MathML with** `\int}^{3}_{0} x \mathrm{d}x`
$\int_0^3 x dx$ <math> <mrow> <msubsup> <mo>&#x0222B;</mo> <mrow> <mn>0</mn> </mrow> <mrow> <mn>3</mn> </mrow> </msubsup> <mi>x</mi> <mi>mathrm</mi> <mrow> <mi>d</mi> </mrow> <mi>x</mi> </mrow> </math>	$\int_0^3 x dx$ <math> <mrow> <msubsup> <mo>&#x0222B;</mo> <mrow> <mn>0</mn> </mrow> <mi>x</mi> </msubsup> <mi>mathrm</mi> <mrow> <mi>d</mi> </mrow> <mi>x</mi> <mrow> <mn>3</mn> </mrow> </mrow> </math>

5.

ANDROID DEVELOPMENT

Tools, Android Architecture, Background Services, Frontend

ANDROID DEVELOPMENT TOOLS

- Language: Java 7
- Tools: Android Studio 2.3
- Device: Android Marshmallow (Samsung)
- Application Framework: AndroidAnnotation
- Libraries
  - **Material Design** Views: MaterialValues, FlexibleAdapter, FloatingActionButtons, MaterialDialog, ProgressActivity
  - LaTeX rendering: MathView and KaTeX
  - Network Access and REST API: Retrofit, RxJava2
  - OCR Related Libraries:
    - Dexter: permission
    - Image Cropper
    - **Image Pre-processing**: Leptonica, Catalano
    - **OCR Engines**: Tesseract, Google Text API

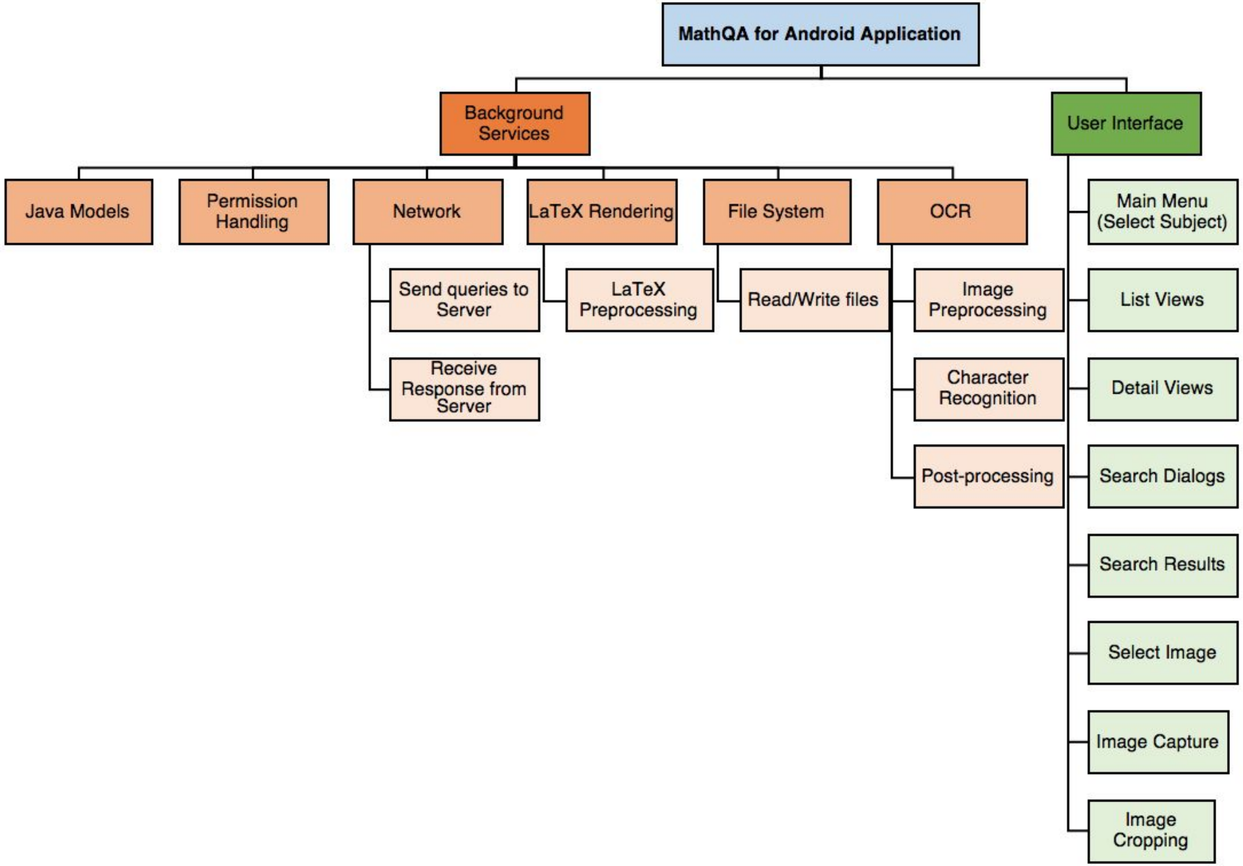
Retrofit

A type-safe **HTTP client** for Android and Java

tesseract-ocr

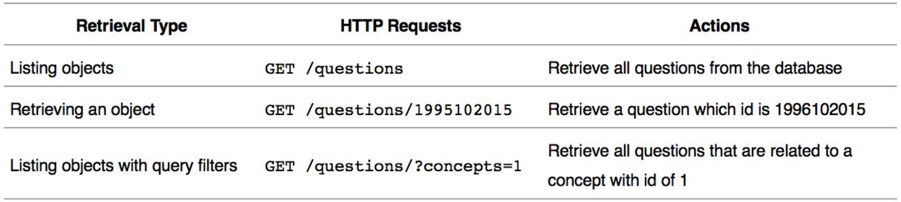
An OCR Engine that was developed at HP Labs between 1985 and 1995... and now at Google.

ANDROID ARCHITECTURE

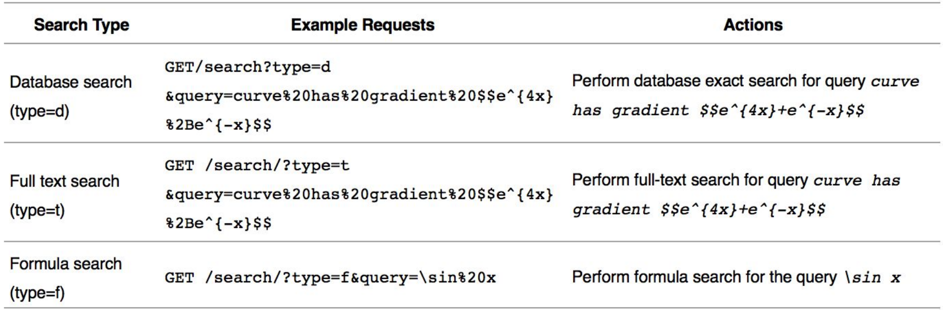


ANDROID DEVELOPMENT - BACKGROUND SERVICES

- **Java object models**: container for storing data from server
- **Network services:**
  - data service: requests data retrieval
  - search service: perform mathematical document retrieval
- **Renders LaTeX:**
  - No LaTeX renderer is compatible with Android ⟶ KaTeX + WebView = **MathView**
  - LaTeX syntax pre-processing:
    - provides alternative view if LaTeX is erroneous or unavailable

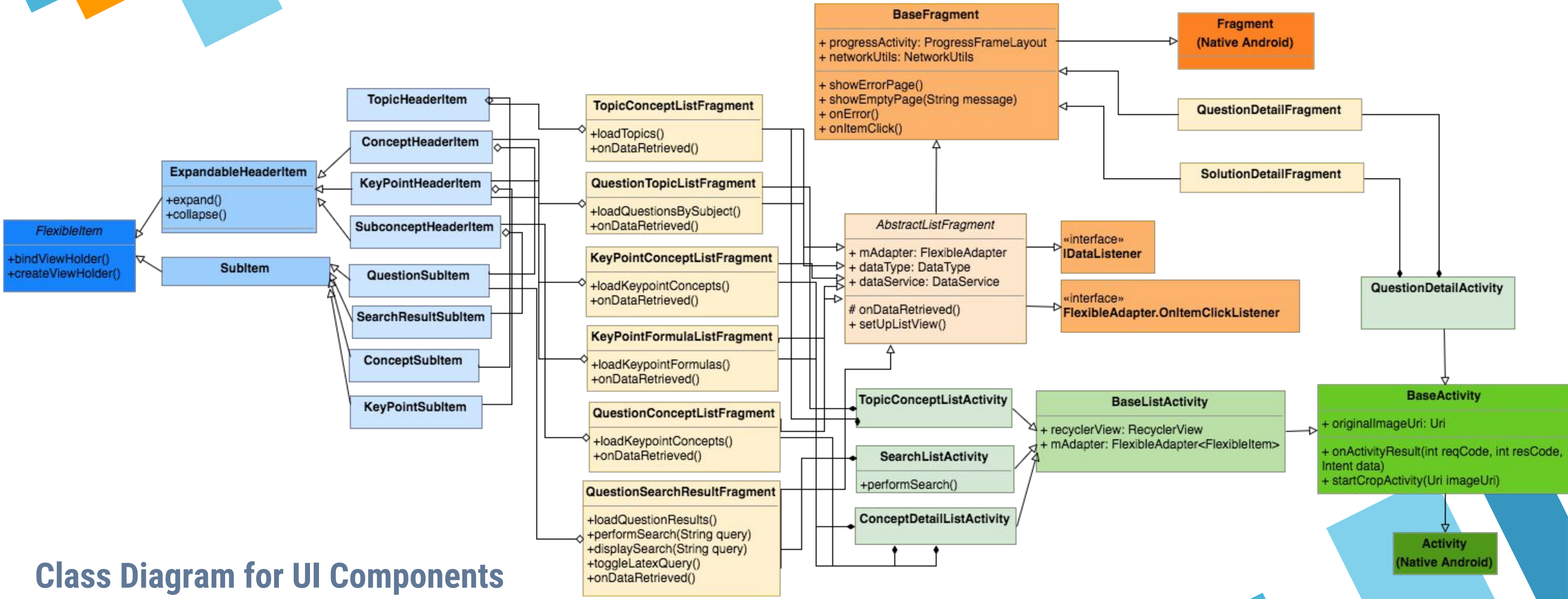


Retrieval Type	HTTP Requests	Actions
Listing objects	`GET /questions`	Retrieve all questions from the database
Retrieving an object	`GET /questions/1995102015`	Retrieve a question which id is 1996102015
Listing objects with query filters	`GET /questions/?concepts=1`	Retrieve all questions that are related to a concept with id of 1



Search Type	Example Requests	Actions
Database search (type=d)	`GET/search?type=d &query=curve%20has%20gradient%20$$e^{4x}%2Be^{-x}$$`	Perform database exact search for query *`curve has gradient $$e^{4x}+e^{-x}$$`*
Full text search (type=t)	`GET /search/?type=t &query=curve%20has%20gradient%20$$e^{4x}%2Be^{-x}$$`	Perform full-text search for query *`curve has gradient $$e^{4x}+e^{-x}$$`*
Formula search (type=f)	`GET /search/?type=f&query=\sin%20x`	Perform formula search for the query *`\sin x`*

ANDROID DEVELOPMENT - FRONTEND



BaseFragment
+ progressActivity: ProgressFrameLayout
+ networkUtils: NetworkUtils
+ showErrorPage()
+ showEmptyPage(String message)
+ onError()
+ onItemClick()

Fragment (Native Android)

QuestionDetailFragment

SolutionDetailFragment

QuestionDetailActivity

AbstractListFragment
+ mAdapter: FlexibleAdapter
+ dataType: DataType
+ dataService: DataService
# onDataRetrieved()
+ setUpListView()

«interface» **IDataListener**

«interface» **FlexibleAdapter.OnItemClickListener**

TopicConceptListFragment
+loadTopics()
+onDataRetrieved()

QuestionTopicListFragment
+loadQuestionsBySubject()
+onDataRetrieved()

KeyPointConceptListFragment
+loadKeypointConcepts()
+onDataRetrieved()

KeyPointFormulaListFragment
+loadKeypointFormulas()
+onDataRetrieved()

QuestionConceptListFragment
+loadKeypointConcepts()
+onDataRetrieved()

QuestionSearchResultFragment
+loadQuestionResults()
+performSearch(String query)
+displaySearch(String query)
+toggleLatexQuery()
+onDataRetrieved()

TopicConceptListActivity

SearchListActivity
+performSearch()

ConceptDetailListActivity

BaseListActivity
+ recyclerView: RecyclerView
+ mAdapter: FlexibleAdapter<FlexibleItem>

BaseActivity
+ originalImageUri: Uri
+ onActivityResult(int reqCode, int resCode, Intent data)
+ startCropActivity(Uri imageUri)

Activity (Native Android)

FlexibleItem
+bindViewHolder()
+createViewHolder()

ExpandableHeaderItem
+expand()
+collapse()

SubItem

TopicHeaderItem

ConceptHeaderItem

KeyPointHeaderItem

SubconceptHeaderItem

QuestionSubItem

SearchResultSubItem

ConceptSubItem

KeyPointSubItem

Class Diagram for UI Components

Commonly occurring view patterns were extracted and abstracted to allow reusable components which include ViewPagerActivities, ListViewFragments, and DetailViewFragments.

6.

OPTICAL CHARACTER RECOGNITION (OCR)

Tools, Workflow, OCR Components, Evaluation, Performance, Discussions

OCR TOOLS

- Language: Java 7
- Tools: Android Studio 2.3
- Device: Android Marshmallow (Samsung)
- Main Libraries:
  - Image Selection and Cropping Tools: Dexter and Image Cropper
  - Image Pre-processing:
    - Leptonica
    - Catalano Framework
  - OCR Engines:
    - Tesseract
    - Google Text API

tesseract-ocr

An OCR Engine that was developed at HP Labs between 1985 and 1995... and now at Google.

OCR WORKFLOW

Steps

1. Obtain the image source: generates image URI

   image picker → access to file system / camera → cropping / rotation

2. OCR pipeline:
   - ◊ many pathways to get OCR result from source image
   - ◊ branching at step 2 and 4 allows a flexibility for OCR processes
   - ◊ Steps:
     - i. Image Source Conversion to Bitmap
     - ii. Bitmap Image Pre-processing
     - iii. OCR on Pre-processed Bitmap

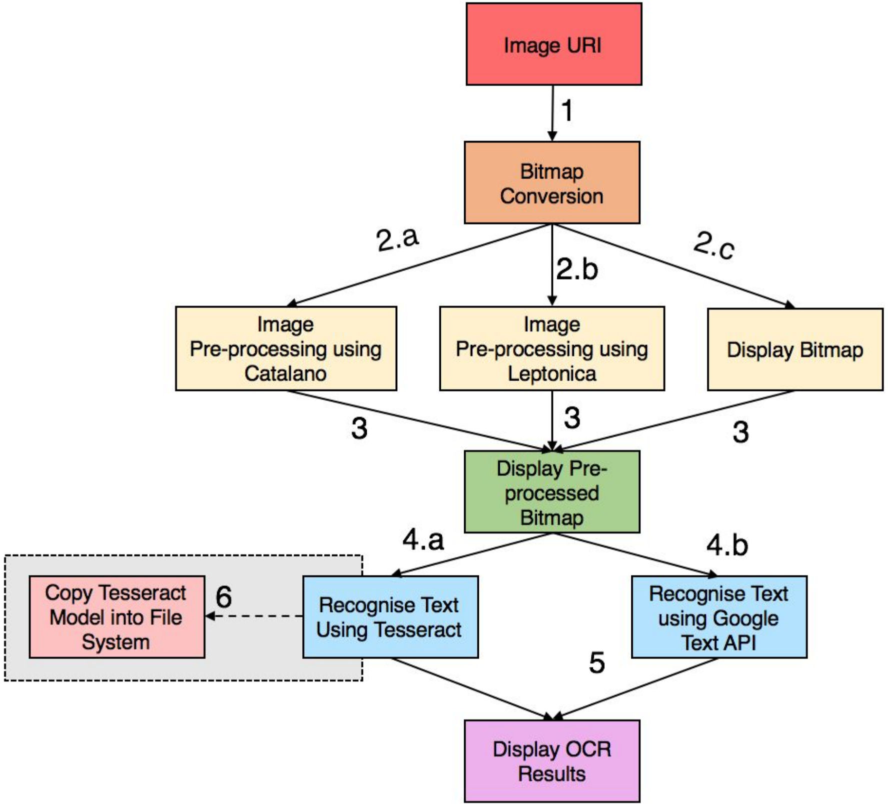


Image URI
1
Bitmap Conversion
2.a
2.b
2.c
Image Pre-processing using Catalano
Image Pre-processing using Leptonica
Display Bitmap
3
3
3
Display Pre-processed Bitmap
4.a
4.b
Copy Tesseract Model into File System
6
Recognise Text Using Tesseract
Recognise Text using Google Text API
5
Display OCR Results

OCR WORKFLOW

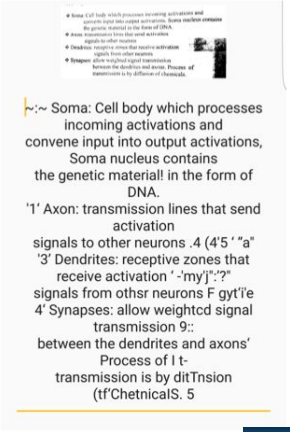
1. **Image Source Conversion to Bitmap**

- Tesseract and Google Text API **only accept bitmap inputs**
- Involves image **downsizing** to produce good quality pre-processed bitmaps

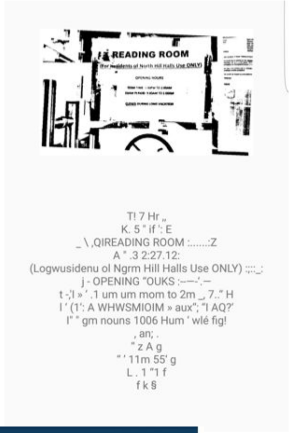
2. **Bitmap Image Pre-processing**

- optional but necessary for Tesseract if image quality is bad
- **conducted experiments** to select best pre-processing actions using Leptonica and Catalano
    - (1) **contrast** and (2) **background normalisation**, (3) **binarisation** using **sauvola's** technique, and (4) image **de-skewing** with Leptonica were shown to have a promising performance

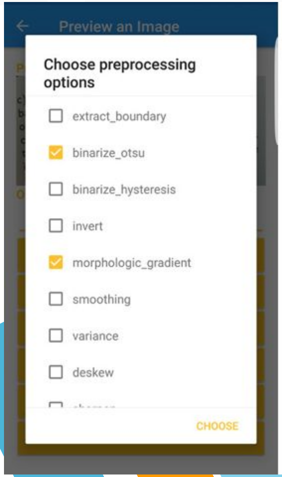
Tesseract Performance Drop



```
~:~ Soma: Cell body which processes
incoming activations and
convene input into output activations,
Soma nucleus contains
the genetic material! in the form of
DNA.
'1' Axon: transmission lines that send
activation
signals to other neurons .4 (4'5 ' "a"
'3' Dendrites: receptive zones that
receive activation ' -'my'j":'?"
signals from othsr neurons F gyt'i'e
4' Synapses: allow weightcd signal
transmission 9::
between the dendrites and axons'
Process of I t-
transmission is by ditTnsion
(tf'ChetnicalS. 5
```



```
T! 7 Hr „
K. 5 " if ': E
_ \ ,QIREADING ROOM :......:Z
A " .3 2:27.12:
(Logwusidenu ol Ngrm Hill Halls Use ONLY) :;::_:
j - OPENING "OUKS :-—-'.—
t -;'l » ' .1 um um mom to 2m _, 7.." H
l ' (1': A WHWSMIOIM » aux"; "I AQ?'
I" " gm nouns 1006 Hum ' wlé fig!
, an; .
" z A g
" ' 11m 55' g
L . 1 "1 f
f k §
```



Choose preprocessing options

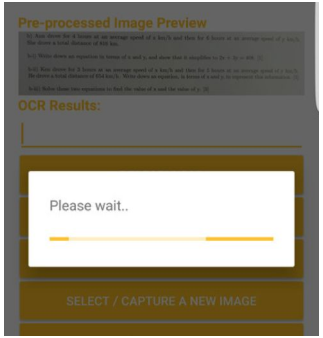
- □ extract_boundary
- ☑ binarize_otsu
- □ binarize_hysteresis
- □ invert
- ☑ morphologic_gradient
- □ smoothing
- □ variance
- □ deskew

OCR WORKFLOW

3. OCR on Pre-processed Bitmap

- pre-processed Bitmap is sent to OCR engine (Tesseract or Google Text API) for character recognition
- OCR result is displayed in an editable text box
- **Best Practices:**
  - OCR is a heavy process
  - blocking mechanism: prevents user interference

Loading (blocking)



Pre-processed Image Preview

b) Ann drove for 4 hours at an average speed of x km/h and then for 6 hours at an average speed of y km/h. She drove a total distance of 816 km.

b-i) Write down an equation in terms of x and y, and show that it simplifies to 2x + 3y = 408. [1]

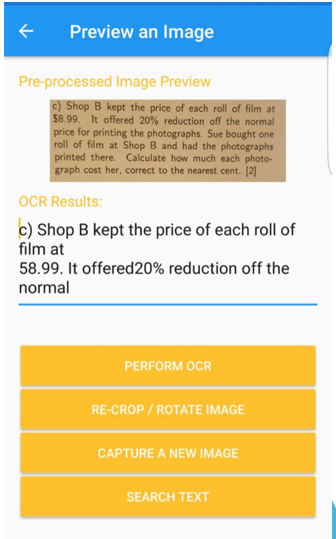
b-ii) Ken drove for 3 hours at an average speed of x km/h and then for 5 hours at an average speed of y km/h. He drove a total distance of 654 km/h. Write down an equation, in terms of x and y, to represent this information. [1]

b-iii) Solve these two equations to find the value of x and the value of y. [3]

OCR Results:

Please wait..

OCR Result



Pre-processed Image Preview

c) Shop B kept the price of each roll of film at $8.99. It offered 20% reduction off the normal price for printing the photographs. Sue bought one roll of film at Shop B and had the photographs printed there. Calculate how much each photograph cost her, correct to the nearest cent. [2]

OCR Results:

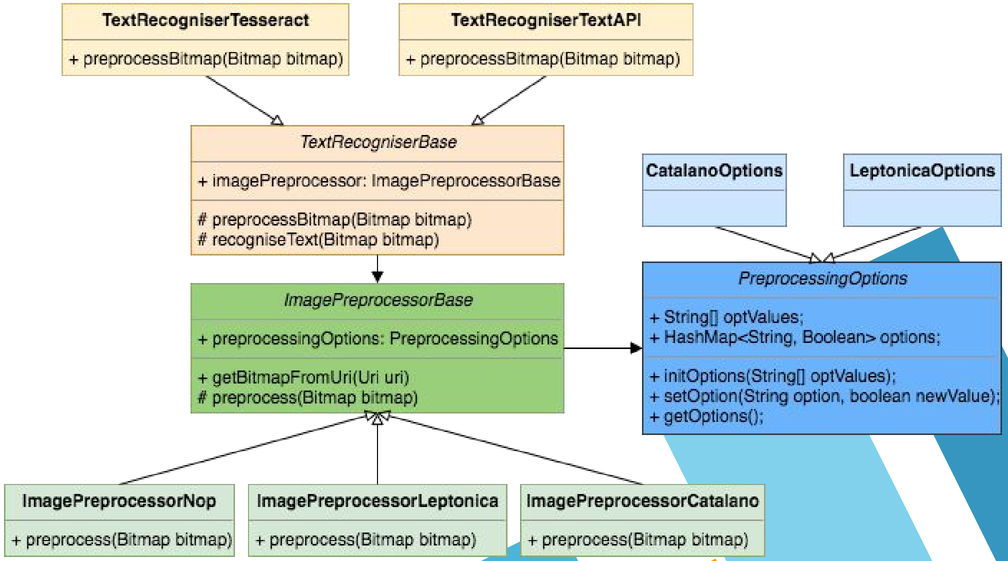
c) Shop B kept the price of each roll of
film at
58.99. It offered20% reduction off the
normal

CLASS DIAGRAM: OCR COMPONENTS

- Common features of OCR processes are abstracted into component base classes

  ⟶ new engines or pre-processors can be added by **extending** from **base classes**

- **Impact**:
  - new engines or image pre-processing tools can be added / modified / removed without affecting other existing components



TextRecogniserTesseract
+ preprocessBitmap(Bitmap bitmap)

TextRecogniserTextAPI
+ preprocessBitmap(Bitmap bitmap)

TextRecogniserBase
+ imagePreprocessor: ImagePreprocessorBase
# preprocessBitmap(Bitmap bitmap)
# recogniseText(Bitmap bitmap)

CatalanoOptions

LeptonicaOptions

ImagePreprocessorBase
+ preprocessingOptions: PreprocessingOptions
+ getBitmapFromUri(Uri uri)
# preprocess(Bitmap bitmap)

PreprocessingOptions
+ String[] optValues;
+ HashMap<String, Boolean> options;
+ initOptions(String[] optValues);
+ setOption(String option, boolean newValue);
+ getOptions();

ImagePreprocessorNop
+ preprocess(Bitmap bitmap)

ImagePreprocessorLeptonica
+ preprocess(Bitmap bitmap)

ImagePreprocessorCatalano
+ preprocess(Bitmap bitmap)

OCR EVALUATION

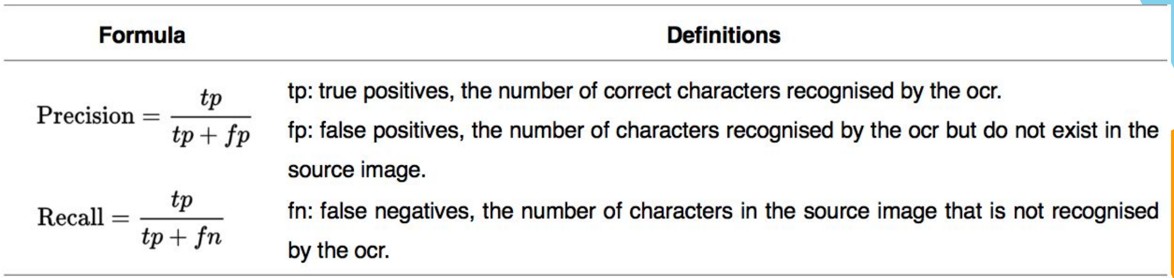
- Experiment carried out to **evaluate** different **OCR engine** and **pre-processor** performance

1. Tesseract without image pre-processing (NOP)
2. Tesseract with image pre-processing using Leptonica
3. Tesseract with image pre-processing using Catalano Framework.
4. Google Text API + NOP

- Tools: MathQA app, Django, Python

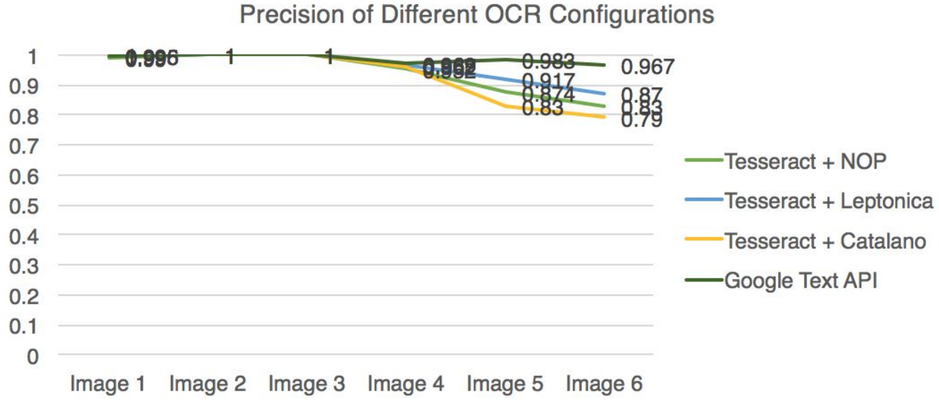
- Test Data Preparation: uses 6 different document images with varying characteristics
  - uses Sans-serif (1-4) / Serif fonts (5-6)
  - contains (4, 6) / does not contain LaTeX (1-3, 5)
  - Skewed (1) vs non-skewed (2-6)
- Evaluation Methodology:

1. Perform OCR on the test image using 4 OCR configurations
2. Evaluate OCR result to calculate the precision, recall, and processing time on each test image for each OCR configuration

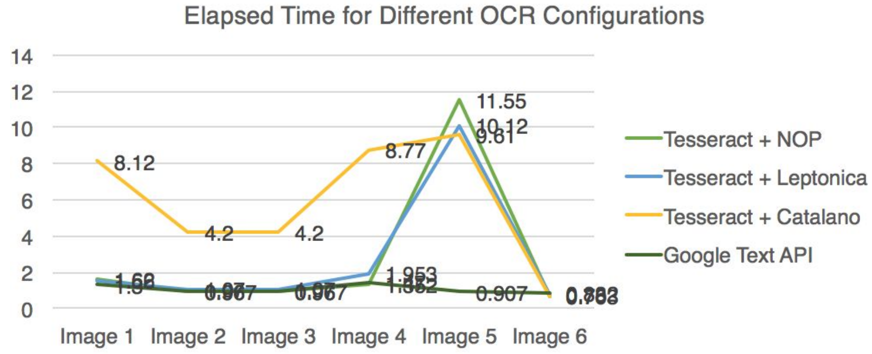


Formula	Definitions
$\text{Precision} = \frac{tp}{tp + fp}$	tp: true positives, the number of correct characters recognised by the ocr. fp: false positives, the number of characters recognised by the ocr but do not exist in the source image.
$\text{Recall} = \frac{tp}{tp + fn}$	fn: false negatives, the number of characters in the source image that is not recognised by the ocr.

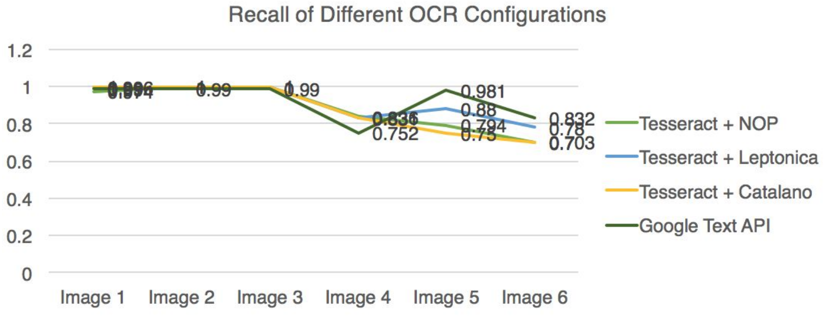
OCR PERFORMANCE



Precision of Different OCR Configurations



Elapsed Time for Different OCR Configurations

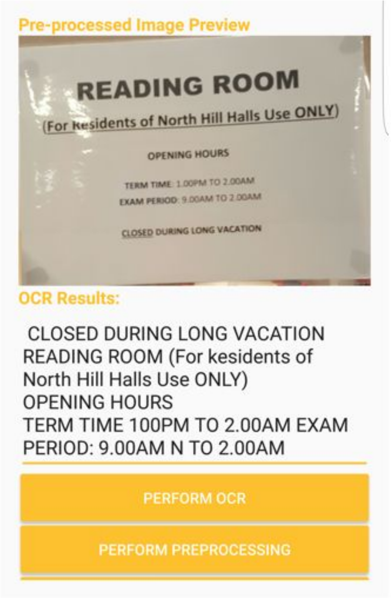


Recall of Different OCR Configurations

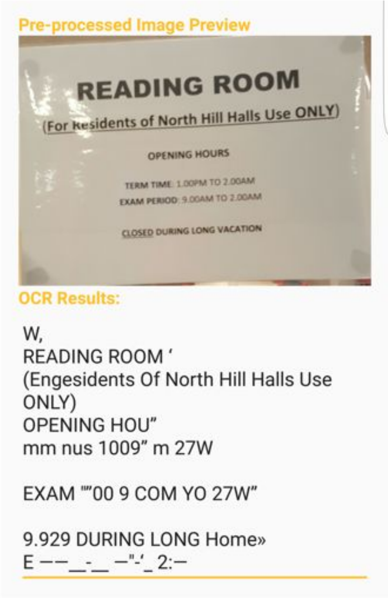
Overall Result

Metrics	Tesseract + NOP	Tesseract + Leptonica	Tesseract + Catalano	Google Text API
Average Precision	0.941	0.96	0.93	0.986
Average Recall	0.884	0.9145	0.88	0.922
Average Time (s)	2.857	2.76	5.93	1.074

OCR PERFORMANCE


Pre-processed Image Preview
READING ROOM
(For kesidents of North Hill Halls Use ONLY)
OPENING HOURS
TERM TIME: 1.00PM TO 2.00AM
EXAM PERIOD: 9.00AM TO 2.00AM
CLOSED DURING LONG VACATION
OCR Results:
PERFORM OCR
PERFORM PREPROCESSING

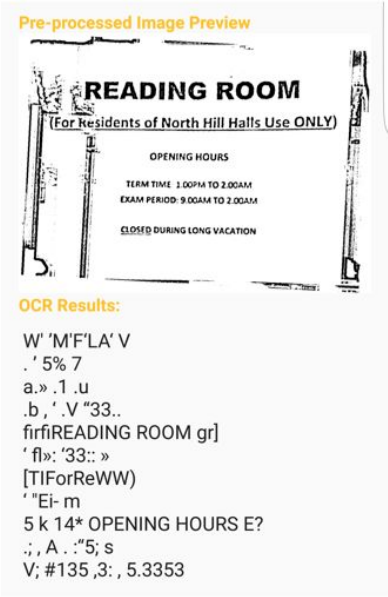
```
 CLOSED DURING LONG VACATION
READING ROOM (For kesidents of
North Hill Halls Use ONLY)
OPENING HOURS
TERM TIME 100PM TO 2.00AM EXAM
PERIOD: 9.00AM N TO 2.00AM
```


Pre-processed Image Preview
READING ROOM
(For kesidents of North Hill Halls Use ONLY)
OPENING HOURS
TERM TIME: 1.00PM TO 2.00AM
EXAM PERIOD: 9.00AM TO 2.00AM
CLOSED DURING LONG VACATION
OCR Results:

```
W,
READING ROOM ‘
(Engesidents Of North Hill Halls Use
ONLY)
OPENING HOU”
mm nus 1009” m 27W

EXAM ""00 9 COM YO 27W”

9.929 DURING LONG Home»
E ——__-__ —"-‘_ 2:—
```


Pre-processed Image Preview
READING ROOM
(For kesidents of North Hill Halls Use ONLY)
OPENING HOURS
TERM TIME 1.00PM TO 2.00AM
EXAM PERIOD: 9.00AM TO 2.00AM
CLOSED DURING LONG VACATION
OCR Results:

```
W' ‘M'F‘LA’ V
. ’ 5% 7
a.» .1 .u
.b , ‘ .V “33..
firfiREADING ROOM gr]
‘ fl»: ‘33:: »
[TIForReWW)
‘ "Ei- m
5 k 14* OPENING HOURS E?
.; , A . :“5; s
V; #135 ,3: , 5.3353
```

Google Text API (left), Tesseract (middle and right) Performance in Recognising a Poor-quality Image

OCR DISCUSSIONS

- When the document image is clean, both Tesseract and Google Text API without image pre-processing are robust
- Catalano can be removed from OCR pipeline
- Tesseract + NOP outperforms all other OCR configurations when image is clean, contain only alphanumerical symbols and use sans-serif fonts
- Tesseract requires training data model to be copied onto the phone
- Google Text API + NOP outperforms all other OCR configurations with the highest precision, recall and fastest processing time

7.

DEMONSTRATION

Android, Admin GUI

MATHQA ANDROID DEMO

https://youtu.be/-ySJp6QnE3w

ADMIN GUI DEMO

https://youtu.be/dE0CVogsEIo

8.

CONCLUSION

Achievements, Conclusion, Future Recommendation

ACHIEVEMENTS

Achievements (App Features)

- **Accessing** mathematical **contents** and perform **mathematical document retrieval**
  - Supports **different modes** of mathematical document **search**
- Viewing Mathematical contents:
  - **LaTeX rendering**
  - **Intuitive** and **logical display** between object models
- **OCR** functionality:
  - **Select document image**: camera capture and file storage access
  - **perform OCR** on the selected image
  - Use OCR results to **find** mathematical documents

Achievements (Design)

- Initial system design plans, architecture and prototypes were meticulously designed to:
  - provide a **maintainable** app that adheres to **good design principles**
  - **sustainable** for **future improvements**
- Incorporated **best design practises** for building good user experience applications as specified in **Material Design** guideline
  - Various tools and libraries that support Material Design were thoroughly **explored** and **highly utilised** throughout the UI development

CONCLUSION

- MathQA system is a web-based learning platform that helps students solve mathematical problems with 2 major tasks:
  - build **database model** and **mathematical document retrieval** services
    - support three mathematical document retrievals: **database**, **full-text**, and **formula-based** retrievals
      - formula-based retrieval uses inverted index technique and evaluated to be **promising** as it has 97.5% MAP@10 and Average P@10 of 91.4%.
  - develop Android application that utilise the services
    - **displays mathematical contents** in **intuitive** manner with Material Design
    - **incorporate OCR** which allows to perform a **retrieval** based on **image capture**
      - **Google Text API** is found to be the best performing among all engines
- Software engineering principles were adhered to produce software that is **maintainable**, **extensible** and **sustainable** for future developers

→ **Project objective** is successfully **achieved**

FUTURE RECOMMENDATION

- Mathematical Document Retrieval:
    - **combine retrieval techniques** for text-based and formula-based instead of executing them separately
    - explore different approaches of **obtaining mathematical structure** from LaTeX
    - Admin GUI can help future developers to **experiment** and **evaluate** new retrieval techniques
- Document Image Recognition
    - extend OCR to **recognise LaTeX** content from image
- Extending MathQA Features
    - enhance existing features by developing **dynamic** and **interactive** learning platform where users can **interact** with their peers, teachers via **discussions**, **QA** on top of the current app
- Conduct Usability Study
    - determine **users' satisfaction** and areas to be improved

THANK YOU :)

Any questions?

BIBLIOGRAPHY

[1] M. Kai, “Data Mining for Mathematical Question Answering Community,” Singapore, 2011

[2] S. H. Samarasinghe, “Semantic-based Retrieval for Mathematical Knowledge,” Singapore, 2009

[3] A. A. Deka, “Web-based Mathematical Document Retrieval for Mobile Android Application”, Singapore, 2017