Project of Master of Computing
and Innovation

AI driven tool for supporting software developers in addressing security challenges

GROUP：OS1E

Ruotong Chang	a1873476
Chung-Ju Lin	a1906146
Yuan Lai	a1879130
Wei Ding	a1895110
Mingkun Ma	a1897693

CONTENT

01 PROJECT BRIEF

02 TASK 1&2 Topic Categorization & Modeling

03 TASK 3 Challenge and Skill Categorization & Modeling

04 TASK 4 Sentiment analysis of acceptable answers

05 TASK 5 Sentiment & Semantic Similarity Analysis

06 TASK 6 Using LIWC to Compare Linguistic Features of AI vs Human Answers

PART ONE

PROJECT BRIEF

Project Brief

GOAL

Build a 'security problem map' using Gemini AI and NLP.

PROBLEM

Developers face fragmented and poorly classified security information.

SOLUTION

LDA : group questions, challenges and skills
BERT : Sentiment analysis
TfidfVectorizer : cosine similarity
LIWC : Linguistic Features

BENEFIT

Enables faster problem discovery and improves security learning efficiency.

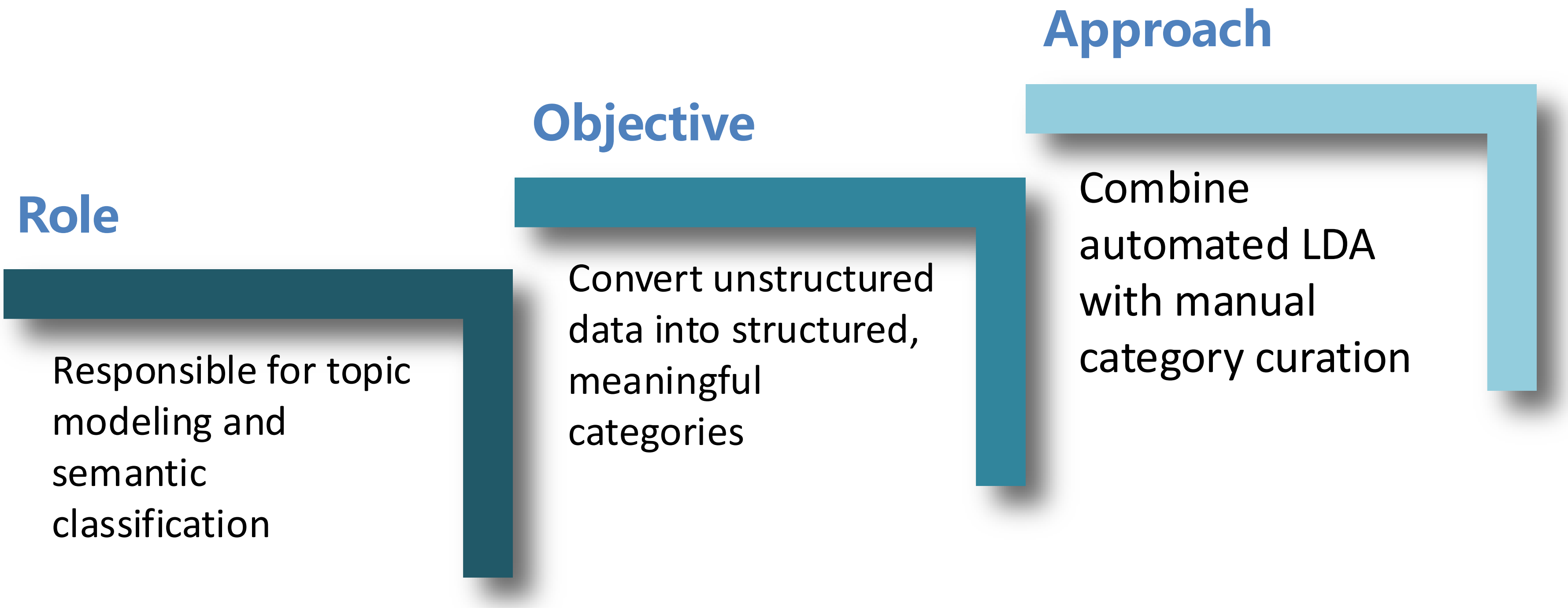
OUTCOME

Clear, scalable structure for learning and solving security problems.

PART TWO

Topic Categorization & Modeling

My Contribution: Topic Categorization & Modeling



Role

Responsible for topic modeling and semantic classification

Objective

Convert unstructured data into structured, meaningful categories

Approach

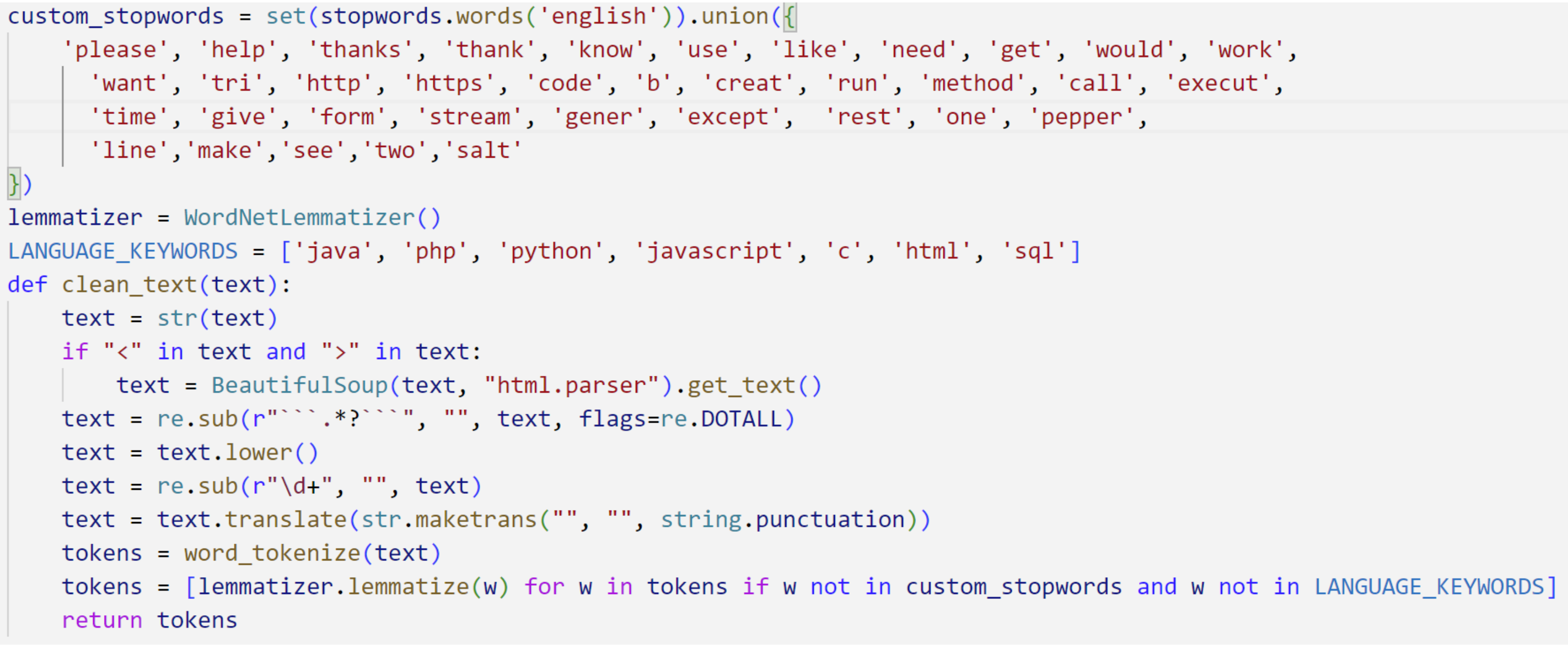
Combine automated LDA with manual category curation

Workflow Overview

1. Data Ingestion: Loaded full Q&A dataset into pandas DataFrame

2. Text Preprocessing:

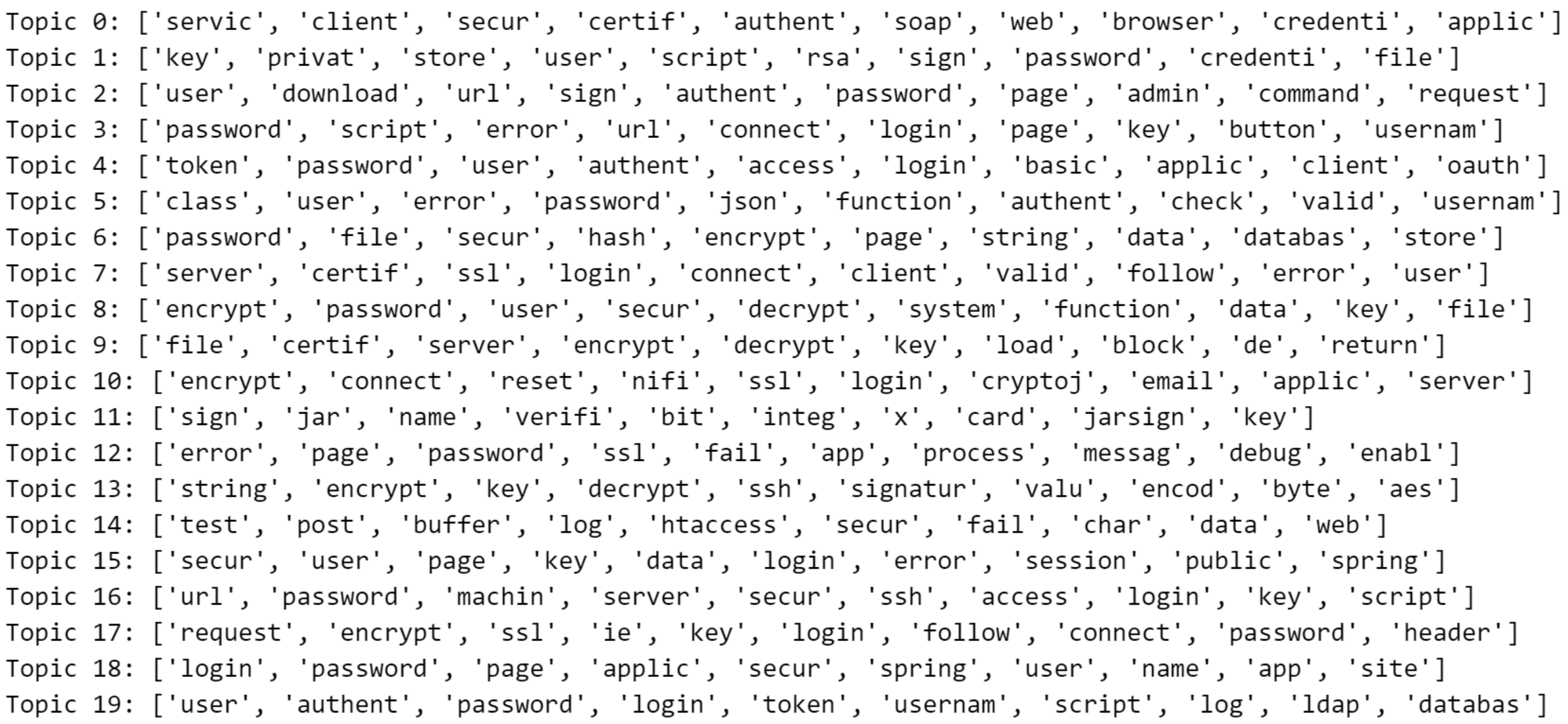
- Lowercased, removed punctuation, stopwords
- Applied lemmatization (NLTK)



````python
custom_stopwords = set(stopwords.words('english')).union({
    'please', 'help', 'thanks', 'thank', 'know', 'use', 'like', 'need', 'get', 'would', 'work',
      'want', 'tri', 'http', 'https', 'code', 'b', 'creat', 'run', 'method', 'call', 'execut',
      'time', 'give', 'form', 'stream', 'gener', 'except',  'rest', 'one', 'pepper',
      'line','make','see','two','salt'
})
lemmatizer = WordNetLemmatizer()
LANGUAGE_KEYWORDS = ['java', 'php', 'python', 'javascript', 'c', 'html', 'sql']
def clean_text(text):
    text = str(text)
    if "<" in text and ">" in text:
        text = BeautifulSoup(text, "html.parser").get_text()
    text = re.sub(r"```.*?```", "", text, flags=re.DOTALL)
    text = text.lower()
    text = re.sub(r"\d+", "", text)
    text = text.translate(str.maketrans("", "", string.punctuation))
    tokens = word_tokenize(text)
    tokens = [lemmatizer.lemmatize(w) for w in tokens if w not in custom_stopwords and w not in LANGUAGE_KEYWORDS]
    return tokens
````

Workflow Overview

3. Topic Modeling: LDA with n_components = 20



```
Topic 0: ['servic', 'client', 'secur', 'certif', 'authent', 'soap', 'web', 'browser', 'credenti', 'applic']
Topic 1: ['key', 'privat', 'store', 'user', 'script', 'rsa', 'sign', 'password', 'credenti', 'file']
Topic 2: ['user', 'download', 'url', 'sign', 'authent', 'password', 'page', 'admin', 'command', 'request']
Topic 3: ['password', 'script', 'error', 'url', 'connect', 'login', 'page', 'key', 'button', 'usernam']
Topic 4: ['token', 'password', 'user', 'authent', 'access', 'login', 'basic', 'applic', 'client', 'oauth']
Topic 5: ['class', 'user', 'error', 'password', 'json', 'function', 'authent', 'check', 'valid', 'usernam']
Topic 6: ['password', 'file', 'secur', 'hash', 'encrypt', 'page', 'string', 'data', 'databas', 'store']
Topic 7: ['server', 'certif', 'ssl', 'login', 'connect', 'client', 'valid', 'follow', 'error', 'user']
Topic 8: ['encrypt', 'password', 'user', 'secur', 'decrypt', 'system', 'function', 'data', 'key', 'file']
Topic 9: ['file', 'certif', 'server', 'encrypt', 'decrypt', 'key', 'load', 'block', 'de', 'return']
Topic 10: ['encrypt', 'connect', 'reset', 'nifi', 'ssl', 'login', 'cryptoj', 'email', 'applic', 'server']
Topic 11: ['sign', 'jar', 'name', 'verifi', 'bit', 'integ', 'x', 'card', 'jarsign', 'key']
Topic 12: ['error', 'page', 'password', 'ssl', 'fail', 'app', 'process', 'messag', 'debug', 'enabl']
Topic 13: ['string', 'encrypt', 'key', 'decrypt', 'ssh', 'signatur', 'valu', 'encod', 'byte', 'aes']
Topic 14: ['test', 'post', 'buffer', 'log', 'htaccess', 'secur', 'fail', 'char', 'data', 'web']
Topic 15: ['secur', 'user', 'page', 'key', 'data', 'login', 'error', 'session', 'public', 'spring']
Topic 16: ['url', 'password', 'machin', 'server', 'secur', 'ssh', 'access', 'login', 'key', 'script']
Topic 17: ['request', 'encrypt', 'ssl', 'ie', 'key', 'login', 'follow', 'connect', 'password', 'header']
Topic 18: ['login', 'password', 'page', 'applic', 'secur', 'spring', 'user', 'name', 'app', 'site']
Topic 19: ['user', 'authent', 'password', 'login', 'token', 'usernam', 'script', 'log', 'ldap', 'databas']
```

→overly narrow or thematically mixed

Workflow Overview

4. Topic Assignment: Assign question to most probable topic and export results for review

- Manual review:

  - Analyzed keywords and example questions

  - Identified recurring technical themes

- Final output: 8 manually curated categories, refined for clarity and technical relevance

Final 8 Categories

final_topic_id	topic_name	topic_description	question_quantity	LDA_topic_id	topic_keywords
1	Identity & Credential Management	authentication protocols, token lifecycles, SSO/STS integration, certificate-based login	71	0	servic, client, secur, certif, authent, soap, web, browser, credenti, applic
				4	token, password, user, authent, access, login, basic, applic, client, oauth
				17	request, encrypt, ssl, ie, key, login, follow, connect, password, header
2	Application Cryptography & Key Management	app-level AES/RSA/HMAC usage, key storage, PKI, salt/derivation, cross-language interoperability	73	1	key, privat, store, user, script, rsa, sign, password, credenti, file
				6	password, file, secur, hash, encrypt, page, string, data, databas, store
				8	encrypt, password, user, secur, decrypt, system, function, data, key, file
3	Web Security: Authn, Session & CSRF/Nonce Integration	web-form login flows, session/cookie issues, CSRF vs. nonce, token-based auth, LDAP bind, dynamic form binding	47	3	password, script, error, url, connect, login, page, key, button, usernam
				12	error, page, password, ssl, fail, app, process, messag, debug, enabl
				14	test, post, buffer, log, htaccess, secur, fail, char, data, web
				15	secur, user, page, key, data, login, error, session, public, spring
4	Cross-Platform Security Integration & Troubleshooting	debugging TLS/SSH/SAML/OAuth across Java/.NET/PHP/JS, smart-card PKCS#11, trust-store problems, multi-platform encryption compatibility	72	7	server, certif, ssl, login, connect, client, valid, follow, error, user
				9	file, certif, server, encrypt, decrypt, key, load, block, de, return
				10	encrypt, connect, reset, nifi, ssl, login, cryptoj, email, applic, server
5	Data-at-Rest Security: File Encryption, Checksums & Hashing	file-system or archive-level encryption, CRC/SHA checksums, zip entry hashing, PEM/P12 private-key protection	49	13	string, encrypt, key, decrypt, ssh, signatur, valu, encod, byte, aes
				16	url, password, machin, server, secur, ssh, access, login, key, script
				18	login, password, page, applic, secur, spring, user, name, app, site
6	Full-Stack Security & End-to-End Integration	front-to-back pipelines (client-side encryption → transport → server validation → DB storage), social-login flows, mixed stacks	20	2	user, download, url, sign, authent, password, page, admin, command, request
				5	class, user, error, password, json, function, authent, check, valid, usernam
7	Common Web-Framework Security Pitfalls	framework-specific gotchas (URL rewrites dropping POST, misconfigured CSRF, Open Redirect, session-sharing, XSS in templating)	29	14	test, post, buffer, log, htaccess, secur, fail, char, data, web
				15	secur, user, page, key, data, login, error, session, public, spring
				16	url, password, machin, server, secur, ssh, access, login, key, script
8	Developer's Practical Crypto & Security Challenges	ad-hoc, language- or tool-specific hurdles (library APIs, IDE integration, CI/CD security scans, REPL debugging)	24	11	sign, jar, name, verifi, bit, integ, x, card, jarsign, key
				19	user, authent, password, login, token, usernam, script, log, ldap, databas

Reflection & Impact

- LDA helped discover latent patterns
- Manual curation improved semantic clarity
- Final structure enables:
  - Improved downstream classification and visualization
  - Clearer communication of developer security challenges
  - Stronger alignment with real-world development issues

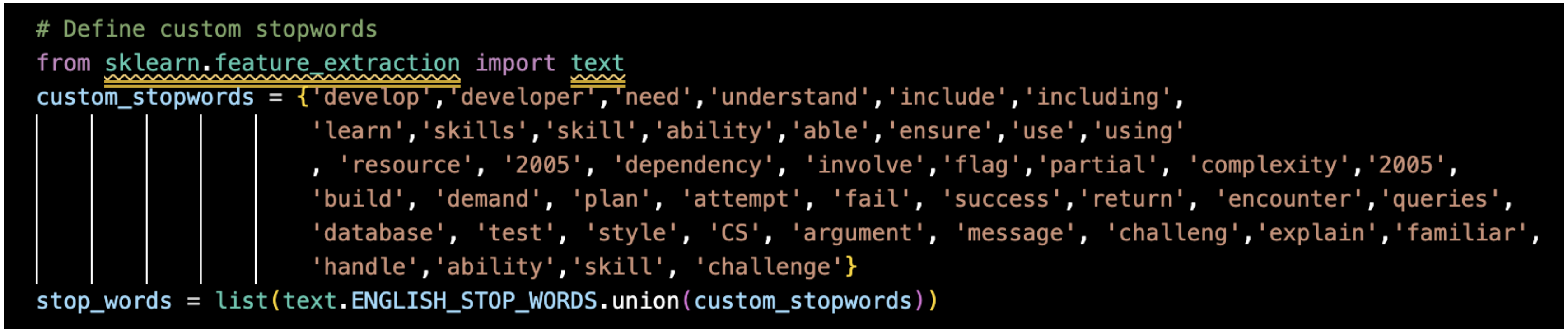
PART THREE

Challenge and Skill Categorization & Modeling

Process Workflow

workflow	Description
Text Preprocessing	Word segmentation, stop word removal, TF-IDF vectorization
Model Training	LAD model to catch the keywords in each topic SVM model to predict each text into one topic
Performance Evaluation	Testing Precision, Recall, F1 score in SVM classification
Manual Verification	Compare the predict topic and the original text we collect.

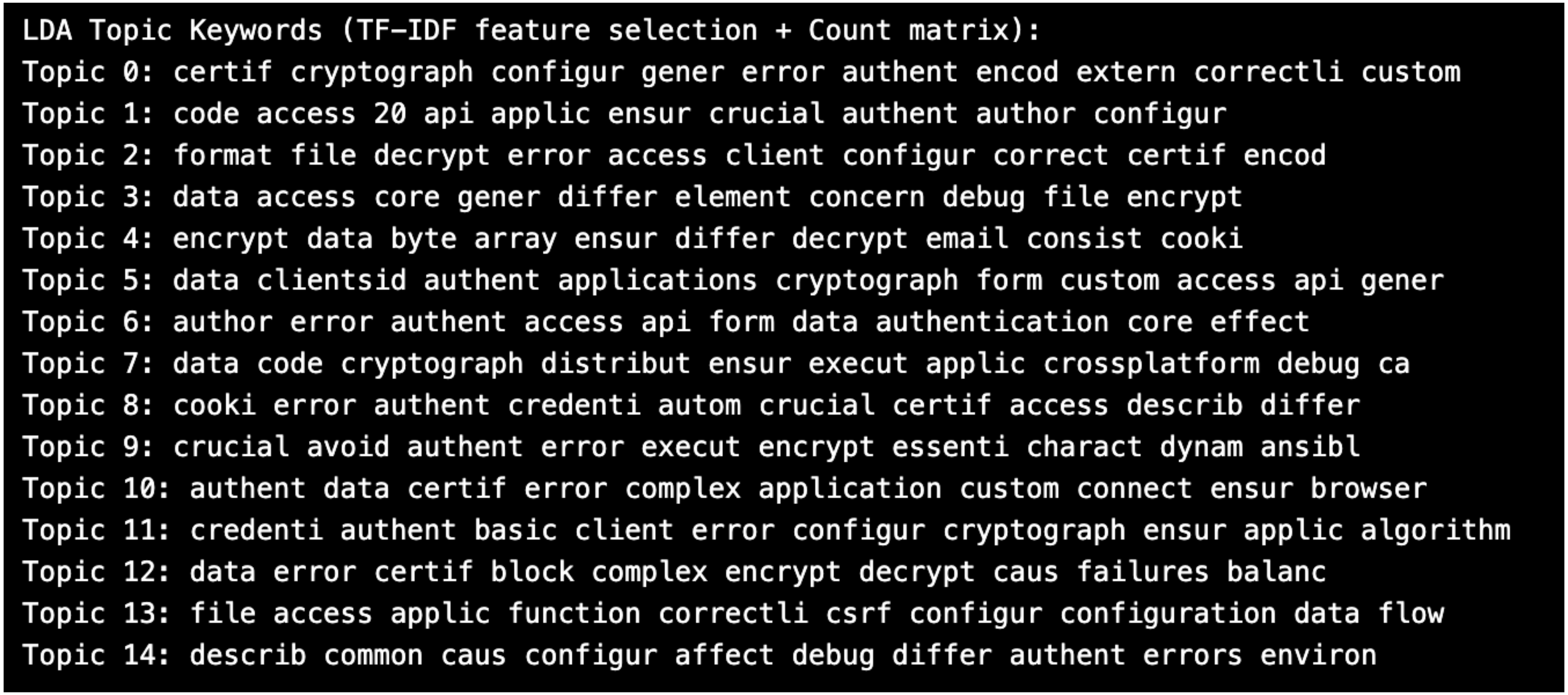
Testing & Refinement



```
# Define custom stopwords
from sklearn.feature_extraction import text
custom_stopwords = {'develop','developer','need','understand','include','including',
                    'learn','skills','skill','ability','able','ensure','use','using'
                    , 'resource', '2005', 'dependency', 'involve','flag','partial', 'complexity','2005',
                    'build', 'demand', 'plan', 'attempt', 'fail', 'success','return', 'encounter','queries',
                    'database', 'test', 'style', 'CS', 'argument', 'message', 'challeng','explain','familiar',
                    'handle','ability','skill', 'challenge'}
stop_words = list(text.ENGLISH_STOP_WORDS.union(custom_stopwords))
```

Testing & Refinement

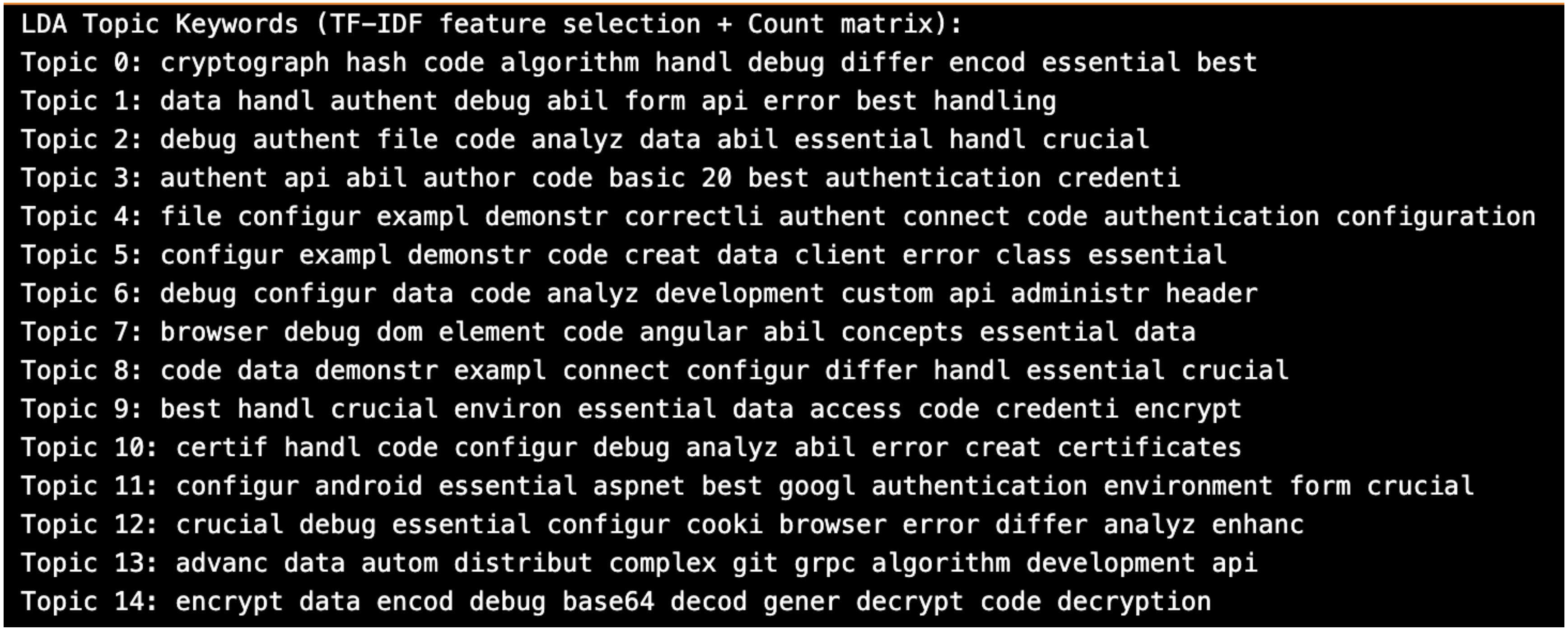
The keyword from challenge:



```
LDA Topic Keywords (TF-IDF feature selection + Count matrix):
Topic 0: certif cryptograph configur gener error authent encod extern correctli custom
Topic 1: code access 20 api applic ensur crucial authent author configur
Topic 2: format file decrypt error access client configur correct certif encod
Topic 3: data access core gener differ element concern debug file encrypt
Topic 4: encrypt data byte array ensur differ decrypt email consist cooki
Topic 5: data clientsid authent applications cryptograph form custom access api gener
Topic 6: author error authent access api form data authentication core effect
Topic 7: data code cryptograph distribut ensur execut applic crossplatform debug ca
Topic 8: cooki error authent credenti autom crucial certif access describ differ
Topic 9: crucial avoid authent error execut encrypt essenti charact dynam ansibl
Topic 10: authent data certif error complex application custom connect ensur browser
Topic 11: credenti authent basic client error configur cryptograph ensur applic algorithm
Topic 12: data error certif block complex encrypt decrypt caus failures balanc
Topic 13: file access applic function correctli csrf configur configuration data flow
Topic 14: describ common caus configur affect debug differ authent errors environ
```

Testing & Refinement

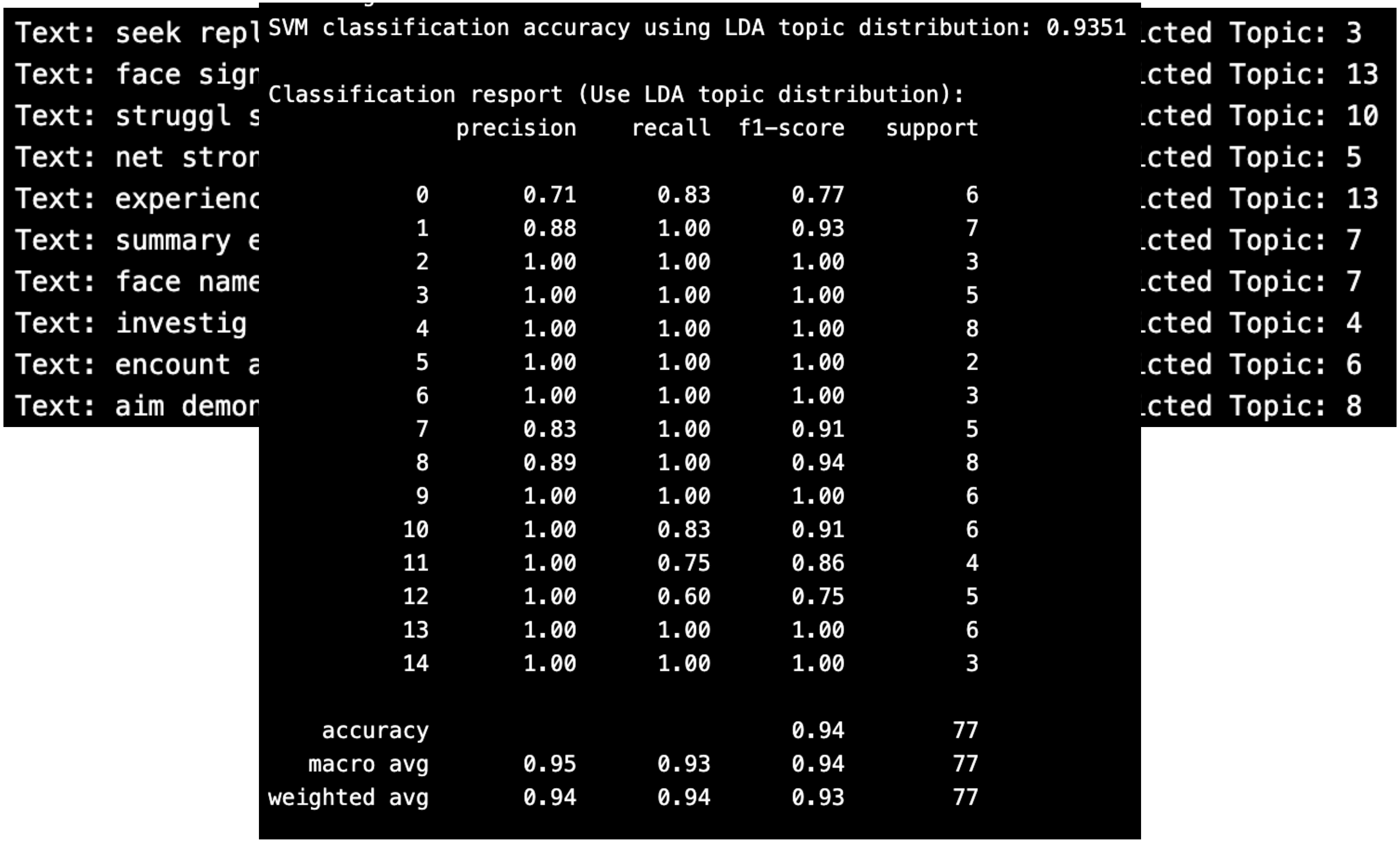
The keyword from skill:



```
LDA Topic Keywords (TF-IDF feature selection + Count matrix):
Topic 0: cryptograph hash code algorithm handl debug differ encod essential best
Topic 1: data handl authent debug abil form api error best handling
Topic 2: debug authent file code analyz data abil essential handl crucial
Topic 3: authent api abil author code basic 20 best authentication credenti
Topic 4: file configur exampl demonstr correctli authent connect code authentication configuration
Topic 5: configur exampl demonstr code creat data client error class essential
Topic 6: debug configur data code analyz development custom api administr header
Topic 7: browser debug dom element code angular abil concepts essential data
Topic 8: code data demonstr exampl connect configur differ handl essential crucial
Topic 9: best handl crucial environ essential data access code credenti encrypt
Topic 10: certif handl code configur debug analyz abil error creat certificates
Topic 11: configur android essential aspnet best googl authentication environment form crucial
Topic 12: crucial debug essential configur cooki browser error differ analyz enhanc
Topic 13: advanc data autom distribut complex git grpc algorithm development api
Topic 14: encrypt data encod debug base64 decod gener decrypt code decryption
```

Testing & Refinement

SVM model to classified the text



```
Text: seek repl                                              .cted Topic: 3
Text: face sigr                                              .cted Topic: 13
Text: struggl s                                              .cted Topic: 10
Text: net stror                                              .cted Topic: 5
Text: experienc                                              .cted Topic: 13
Text: summary e                                              .cted Topic: 7
Text: face name                                              .cted Topic: 7
Text: investig                                               .cted Topic: 4
Text: encount a                                              .cted Topic: 6
Text: aim demor                                              .cted Topic: 8
```

```
SVM classification accuracy using LDA topic distribution: 0.9351

Classification resport (Use LDA topic distribution):
              precision    recall  f1-score   support

           0       0.71      0.83      0.77         6
           1       0.88      1.00      0.93         7
           2       1.00      1.00      1.00         3
           3       1.00      1.00      1.00         5
           4       1.00      1.00      1.00         8
           5       1.00      1.00      1.00         2
           6       1.00      1.00      1.00         3
           7       0.83      1.00      0.91         5
           8       0.89      1.00      0.94         8
           9       1.00      1.00      1.00         6
          10       1.00      0.83      0.91         6
          11       1.00      0.75      0.86         4
          12       1.00      0.60      0.75         5
          13       1.00      1.00      1.00         6
          14       1.00      1.00      1.00         3

    accuracy                           0.94        77
   macro avg       0.95      0.93      0.94        77
weighted avg       0.94      0.94      0.93        77
```

Challenge Categories

Topic	Keywords	Final Name
0	certif, cryptograph, configur, error, authent	Certificate Configuration and Cryptography Errors
1	code, access, api, applic, authent	API Authentication and Authorization Application
2	format, file, decrypt, error, access	File Access and Certificate Decryption Errors
3	data, access, debug, encrypt, core	Data Access and Encryption Debugging
4	encrypt, consistency, decrypt, byte, aes	Encryption Consistency and Decryption
5	data, clientsid, authent, applic, cryptograph	Client Authentication and Cryptography in Applications
6	author, authent, access, api, form	Authentication Errors and API Access
7	data, cryptograph, crossplatform, debug, ca	Cross-Platform Cryptography and Debugging
8	cooki, authent, credenti, error, access	Cookie Authentication and Credential Errors
9	crucial, authent, encrypt, dynamic, manag	Dynamic Authentication and Encryption Management
10	authent, data, certif, complex, browser	Complex Authentication and Browser Configuration
11	credenti, authent, basic, client, algorithm	Basic Client Authentication and Cryptography
12	certif, error, data, encrypt, failures	Certificate Errors and Data Encryption Failures
13	csrf, configur, data, flow, protection	CSRF Protection and Application Data Flow Configuration
14	describ, caus, configur, authent, environ	Authentication Error Causes and Environment Configuration

Challenge Categories

Group	Topic Number	Description
Certificate & Cryptography Errors	0, 2, 12	Issues around certificate loading/configuration, cryptographic operations, and decryption failures.
Authentication & Authorization	1, 6, 11,14	Concerns OAuth/JWT/API-key flows, token-based and basic authentication/authorization processes.
Cookie & Session Management	8	Focused on browser cookie handling, session tokens, and related credential errors.
Cross-Platform & Client-Side Cryptography	5, 7	Multi-language/platform compatibility and debugging for client authentication and cryptographic libraries.
Data Encryption & Debugging	3, 4	Covers data-access encryption pipelines (e.g. AES/RSA), consistency checks, and troubleshooting encryption/decryption.
Dynamic Authentication & Encryption Management	9	End-to-end strategies for dynamically configuring and managing both authentication and encryption settings.
Complex Authentication & Browser Configuration	10	Advanced auth flows, complex setup scenarios, and browser-specific security/policy considerations.
CSRF Protection & Data Flow Configuration	13	Techniques for preventing CSRF attacks and securely configuring application data flows.

Skill Categories

Topic	Top Keywords	Final Name
0	cryptograph, hash, code, algorithm, handle, debug	Cryptographic Algorithms and Debugging
1	data, handle, authent, debug, API, error, best	Authentication and API Error Handling
2	debug, authent, file, code, analyze, data, handle	Debugging Authentication and File Handling
3	authent, API, credential, basic, authentication	API Authentication and Credential Handling
4	file, configur, demonstr, authent, connect, code	File Configuration and Authentication Setup
5	configur, demonstr, code, data, client, error	Data Configuration and Client Error Handling
6	debug, configur, data, code, development, header	Spring and Enterprise Integration Debugging
7	browser, debug, DOM, Angular, concepts	Browser Debugging and Angular Concepts
8	code, data, demonstr, connect, configur, handle	Code Data Configuration and Handling Techniques
9	data, access, credential, encrypt, environment	Data Access and Credential Encryption
10	certif, handle, code, configur, debug, certificates	Certificate Handling and Configuration
11	configur, Android, ASP.NET, Google, authentication	Cross-platform Authentication Configuration
12	debug, configur, cookie, browser, error, analysis	Browser Cookie Debugging and Error Analysis
13	advanced, data, automation, algorithm, development	Advanced Data Automation and Algorithm Development
14	encrypt, data, encode, debug, decode, decrypt	Data Encryption and Decryption

Skill Categories

Group	Topic Number	Description
Cryptography & Encryption	0, 9, 10, 14	Covers low-level crypto operations, certificate handling, encryption/decryption pipelines and debugging.
Authentication & Authorization	1, 3, 11	Focuses on identity workflows, API token flows, credential handling and cross-platform auth configuration.
File & Data Configuration	2, 4, 5, 8	Deals with file access, decryption errors, data configuration and client-side error handling techniques.
Framework & Browser Debugging	6, 7, 12	Involves debugging in Spring/enterprise frameworks, Angular/browser environments and cookie errors.
Automation & Advanced Algorithms	13	Encompasses advanced data automation, complex algorithm development and distributed processing.

Reflection & Impact

- Manual testing can increase accuracy for this task

- The following goals can be achieved:
  - It is more convenient to view relevant information
  - Understand the challenges and skills behind the problem
  - Gemini provides more comprehensive information

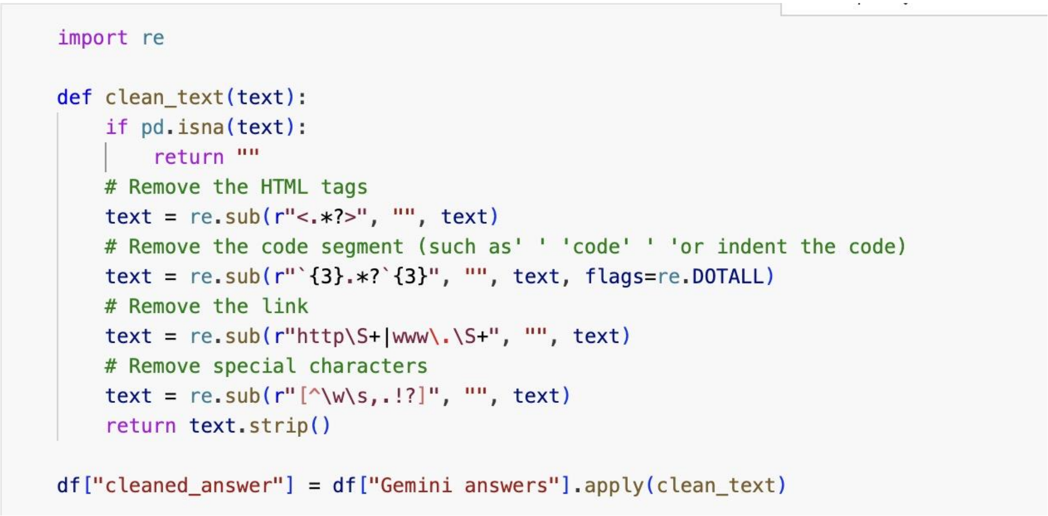
PART FOUR

Sentiment analysis of acceptable answers

Research method

Data reading and preprocessing

First, use pandas to read the data containing questions and answers from the Excel file. To improve the accuracy of the model, I wrote the clean_text() function, which uses regular expressions to remove HTML tags, code blocks, website links and special characters, and only retains the natural language text that is meaningful for sentiment judgment.



```
import re

def clean_text(text):
    if pd.isna(text):
        return ""
    # Remove the HTML tags
    text = re.sub(r"<.*?>", "", text)
    # Remove the code segment (such as' ' 'code' ' 'or indent the code)
    text = re.sub(r"`{3}.*?`{3}", "", text, flags=re.DOTALL)
    # Remove the link
    text = re.sub(r"http\S+|www\.\S+", "", text)
    # Remove special characters
    text = re.sub(r"[^\w\s,.!?]", "", text)
    return text.strip()

df["cleaned_answer"] = df["Gemini answers"].apply(clean_text)
```

Reserch method

BERT Sentiment Analysis Model

Using the pipeline("sentiment-analysis") interface in the transformers library provided by Hugging Face, a fine-tuned BERT model (such as DistilBERT) is called to classify the text sentiment. The model outputs a label (such as POSITIVE, NEGATIVE, NEUTRAL) and the corresponding confidence score (score) for each text segment.

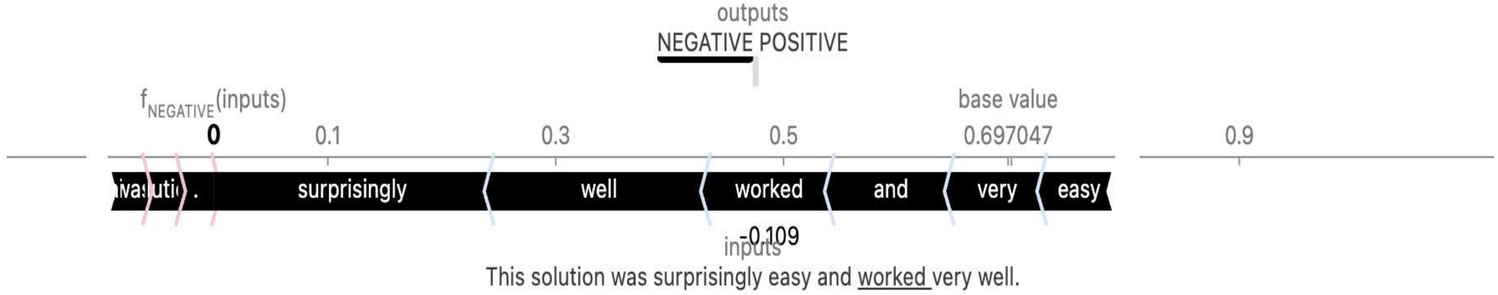
Statistics and Visualization

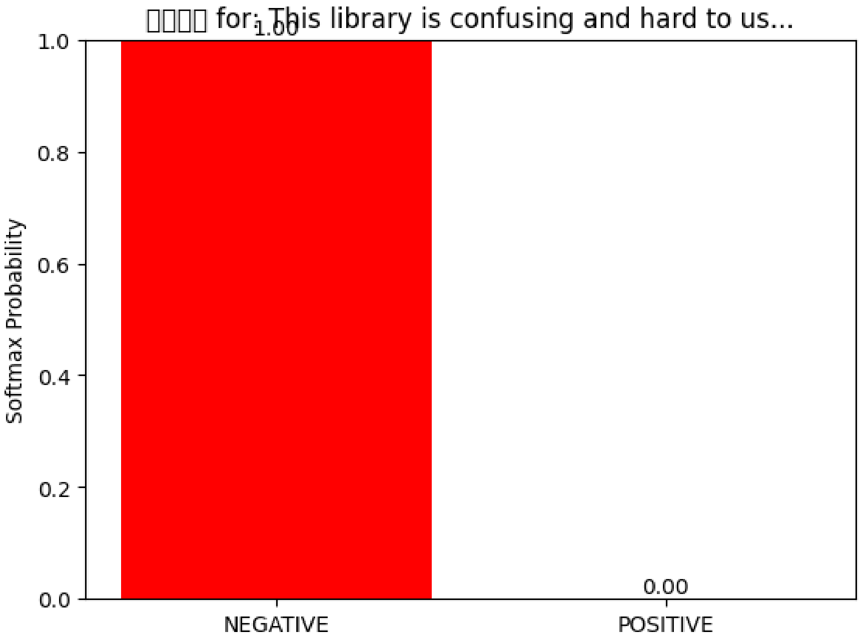
The classification results were visualized using matplotlib, and the number of emotion types was plotted as bar charts and pie charts. The chart clearly shows the distribution differences of each emotion in the two sources.

The working principle of Bert

# Sample Text

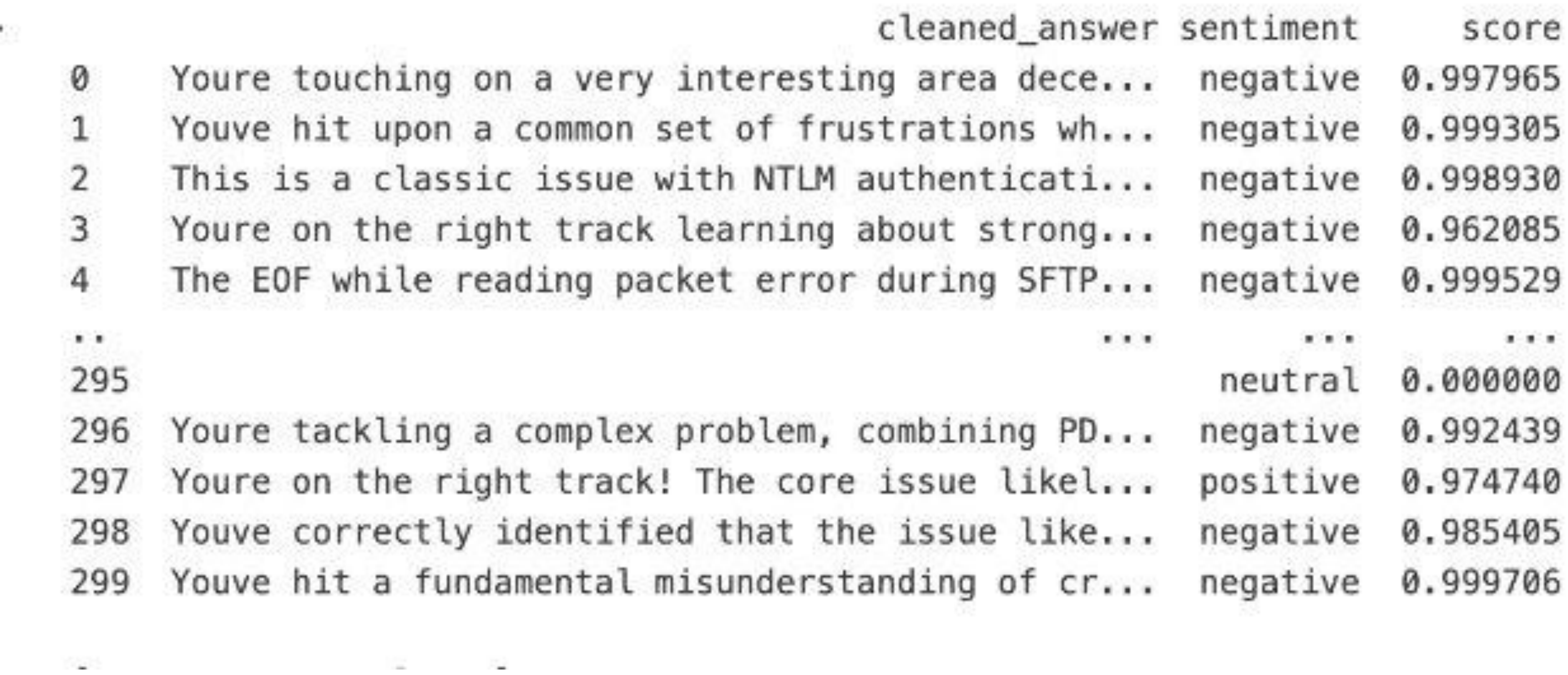
"This solution was surprisingly easy and worked very well."





The working principle of Bert

Present the results during the analysis process



```
                                        cleaned_answer sentiment     score
0    Youre touching on a very interesting area dece...  negative  0.997965
1    Youve hit upon a common set of frustrations wh...  negative  0.999305
2    This is a classic issue with NTLM authenticati...  negative  0.998930
3    Youre on the right track learning about strong...  negative  0.962085
4    The EOF while reading packet error during SFTP...  negative  0.999529
..                                                 ...       ...       ...
295                                                      neutral  0.000000
296  Youre tackling a complex problem, combining PD...  negative  0.992439
297  Youre on the right track! The core issue likel...  positive  0.974740
298  Youve correctly identified that the issue like...  negative  0.985405
299  Youve hit a fundamental misunderstanding of cr...  negative  0.999706
```

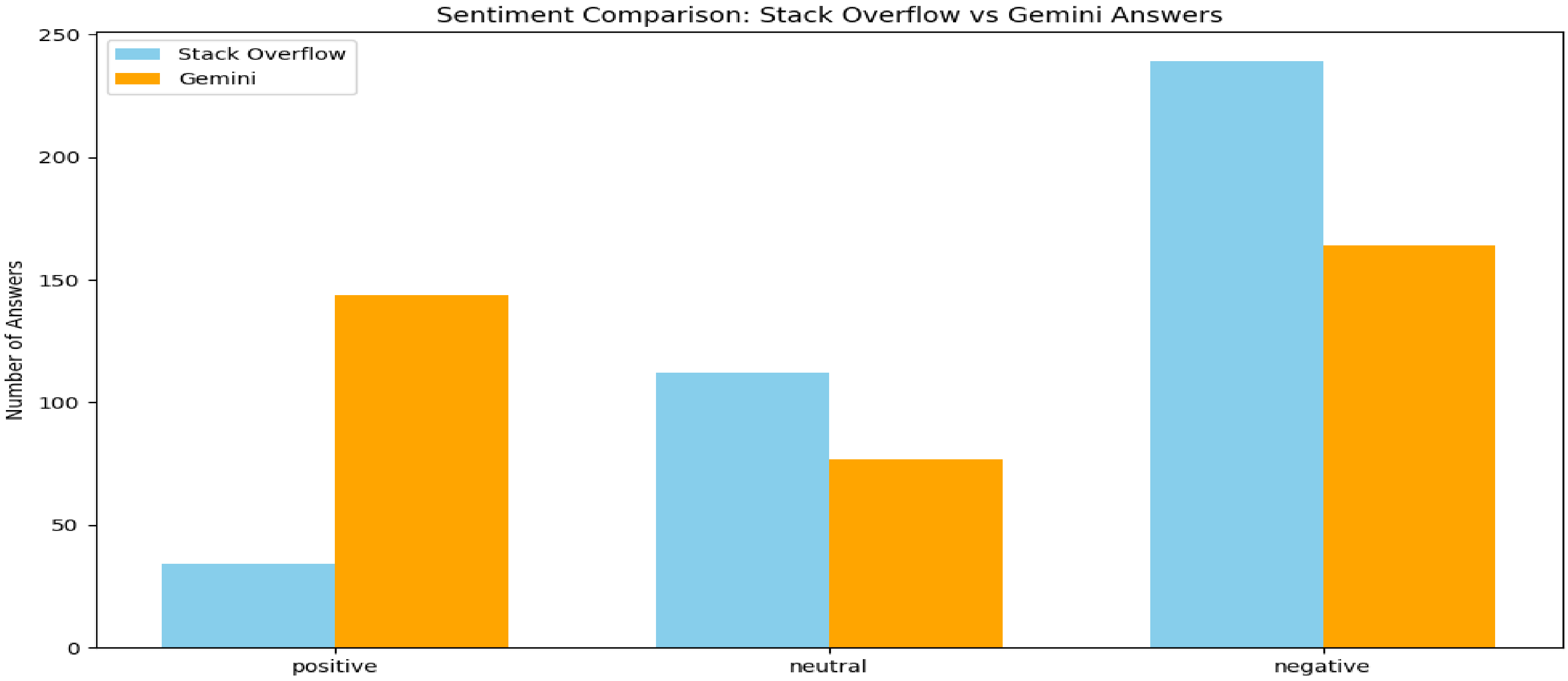
📊 Final results and summary:

Through the chart (as shown in the picture you uploaded), we can intuitively see that:

Gemini's responses are generally more inclined towards positive emotions.

The answers on Stack Overflow are more likely to show negative or neutral.

This might indicate that Gemini tends to offer encouraging and positive responses, while human users may point out problems or express



PART FIVE

Sentiment & Semantic Similarity Analysis

What I Did

Data processing

cleaned & merged data with Task 2 classifications.

Sentiment analysis

calculated polarity & subjectivity using TextBlob.

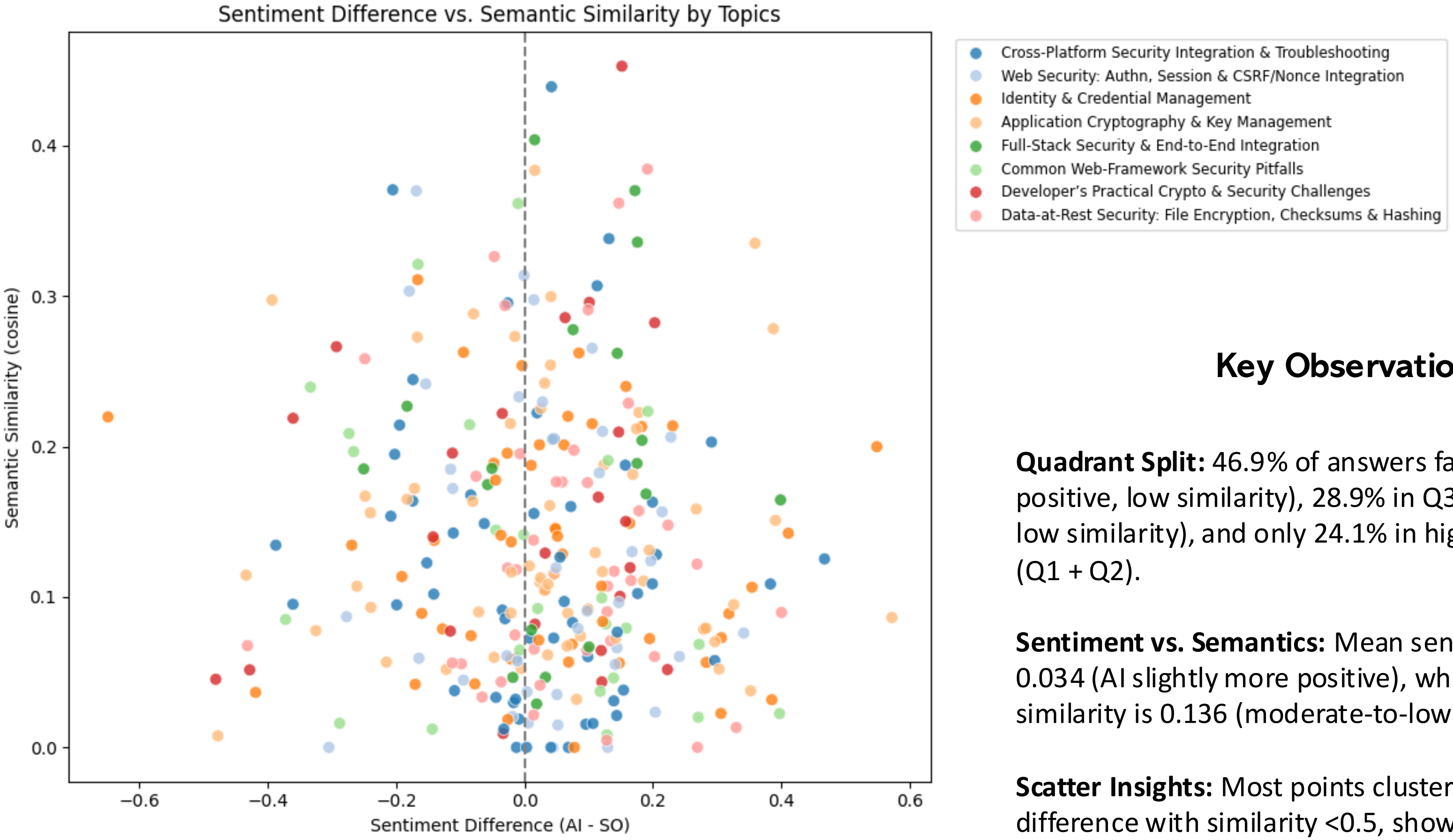
Similarity calculation

TF-IDF vectorization & cosine similarity.

Key Challenge & Solution

Problem: Empty SO answers affected results.
Solution: Filtered out empty answers.

Overall Results



1. **X-axis**: Sentiment Difference (AI – SO)
   - Right side (>0): AI answers more positive
   - Left side (<0): AI answers more negative
2. **Y-axis**: Semantic Similarity (Cosine)
   - Higher position: Higher similarity
   - Lower position: Lower similarity

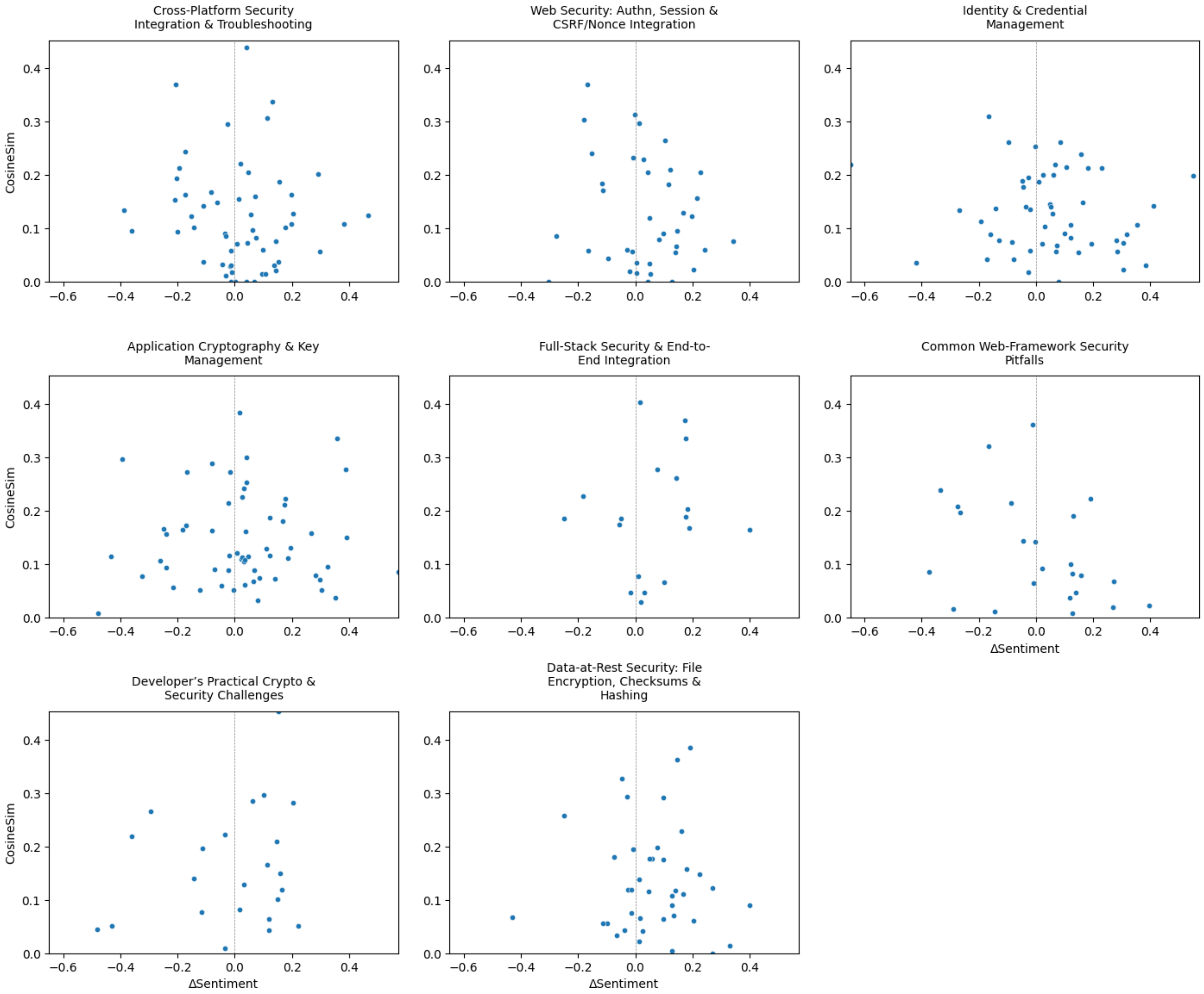
Key Observations:

Quadrant Split: 46.9% of answers fall in Q4 (AI more positive, low similarity), 28.9% in Q3 (SO more positive, low similarity), and only 24.1% in high-similarity quadrants (Q1 + Q2).

Sentiment vs. Semantics: Mean sentiment difference is 0.034 (AI slightly more positive), while mean cosine similarity is 0.136 (moderate-to-low overlap).

Scatter Insights: Most points cluster near zero sentiment difference with similarity <0.5, showing limited phrasing overlap despite occasional alignment.

Overall Results



Key Observations:

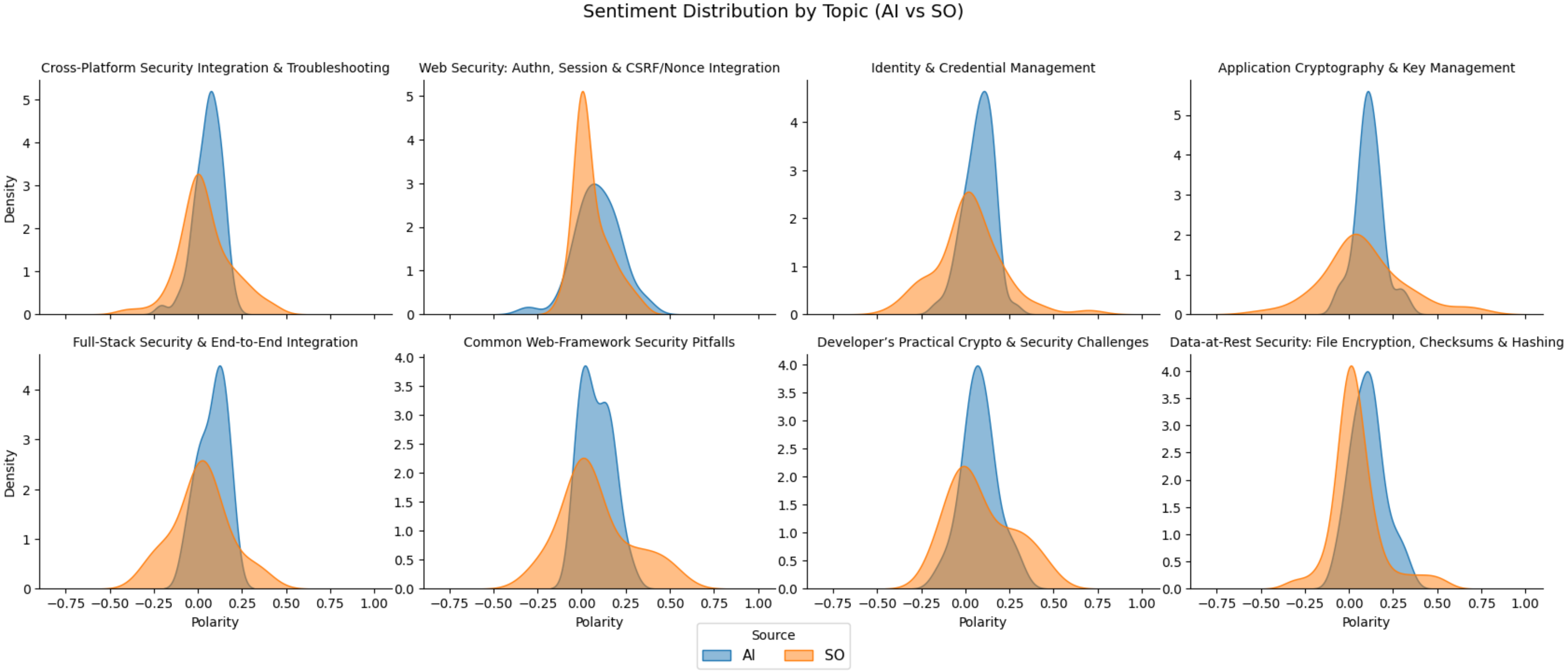
Q1 Ratios: Full-Stack highest (33.3%), Crypto Challenges (21.7%), Web-Framework Pitfalls lowest (4.2%).

Sentiment Gaps: AI is generally more positive; max Δ in Data-at-Rest (0.065) and Full-Stack (0.063), min in Web-Framework (0.003).

Semantic Similarity: Full-Stack peaks at 0.19, Crypto Challenges at 0.159, Cross-Platform lowest at 0.121; most topics ~0.13.

High-Similarity Trends: Except Web-Framework, high-similarity pairs lean AI-positive; 75.8% of all cases lie in low-similarity quadrants.

Detailed Results



Key Observations:

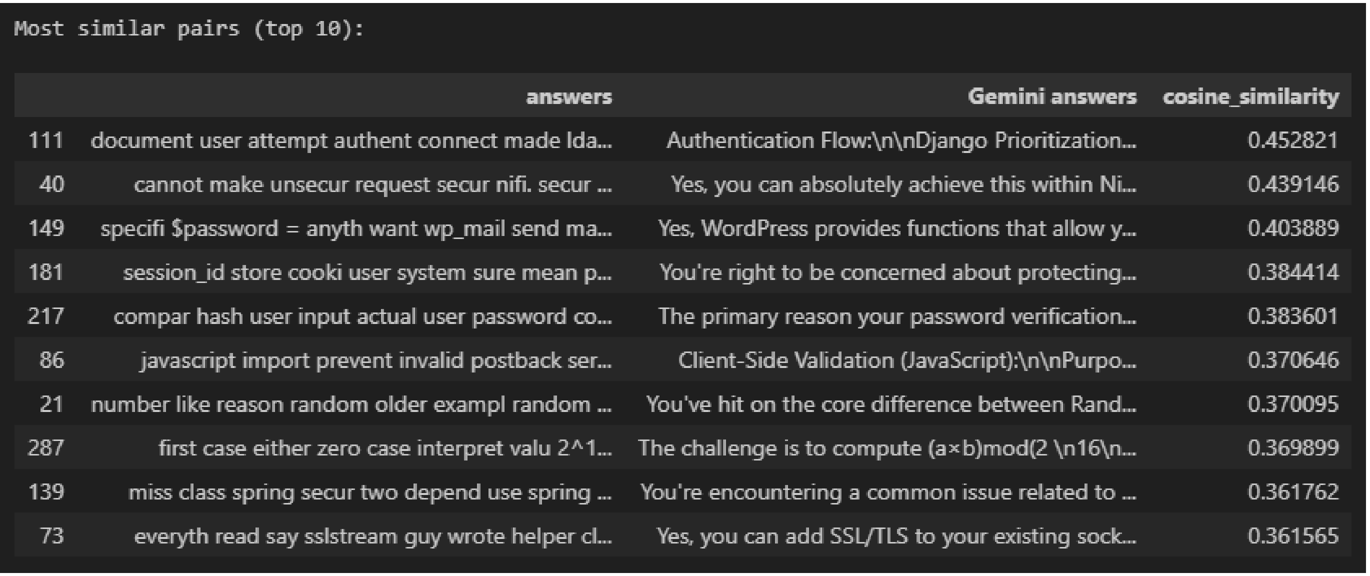
AI Consistency: Across all eight topics, AI's sentiment variance (σ≈0.08–0.13) is consistently lower than humans', showing tighter, more predictable tone.

Small Bias: AI vs. SO polarity differences remain under 0.07 in every topic, indicating overall alignment in positive/negative tone.

Topic Highlights: Largest AI–SO gap in Data-at-Rest (Δ=0.065) and Full-Stack (Δ=0.063); smallest in Web-Framework Pitfalls (Δ=0.003).

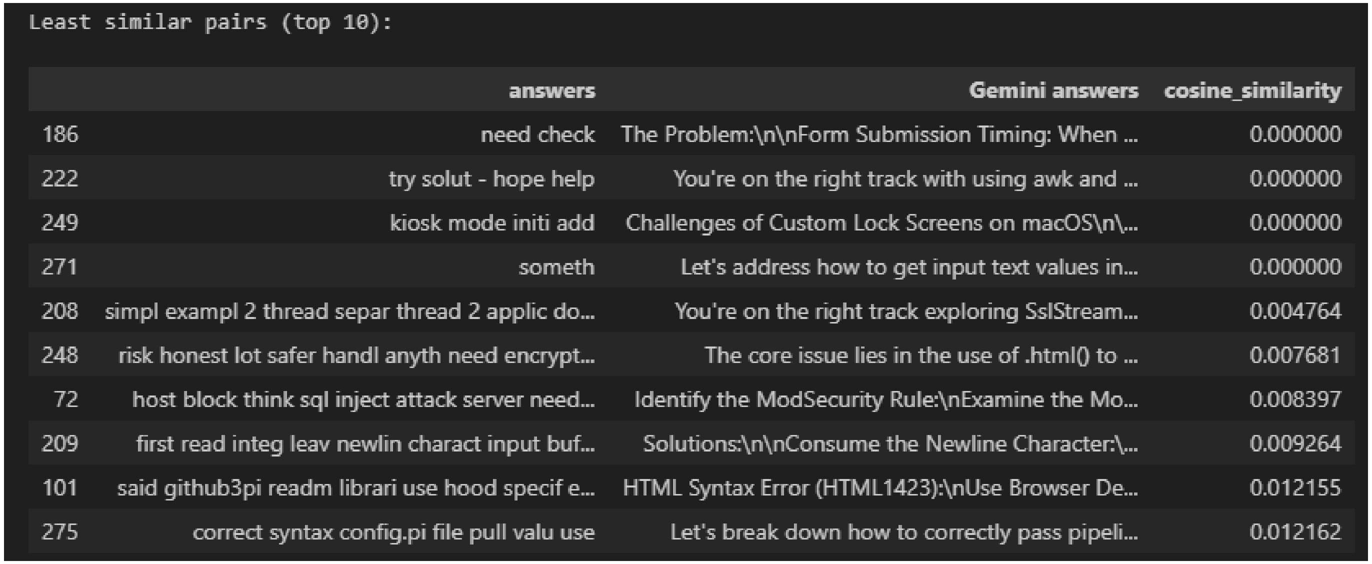
Actionable Insight: Leverage AI's stability for consistent user experience, while human answers add valuable emotional nuance.

Concrete Examples



Most similar pairs (top 10):

	answers	Gemini answers	cosine_similarity
111	document user attempt authent connect made lda...	Authentication Flow:\n\nDjango Prioritization...	0.452821
40	cannot make unsecur request secur nifi. secur ...	Yes, you can absolutely achieve this within Ni...	0.439146
149	specifi $password = anyth want wp_mail send ma...	Yes, WordPress provides functions that allow y...	0.403889
181	session_id store cooki user system sure mean p...	You're right to be concerned about protecting...	0.384414
217	compar hash user input actual user password co...	The primary reason your password verification...	0.383601
86	javascript import prevent invalid postback ser...	Client-Side Validation (JavaScript):\n\nPurpo...	0.370646
21	number like reason random older exampl random ...	You've hit on the core difference between Rand...	0.370095
287	first case either zero case interpret valu 2^1...	The challenge is to compute (a×b)mod(2 \n16\n...	0.369899
139	miss class spring secur two depend use spring ...	You're encountering a common issue related to ...	0.361762
73	everyth read say sslstream guy wrote helper cl...	Yes, you can add SSL/TLS to your existing sock...	0.361565



Least similar pairs (top 10):

	answers	Gemini answers	cosine_similarity
186	need check	The Problem:\n\nForm Submission Timing: When ...	0.000000
222	try solut - hope help	You're on the right track with using awk and ...	0.000000
249	kiosk mode initi add	Challenges of Custom Lock Screens on macOS\n\...	0.000000
271	someth	Let's address how to get input text values in...	0.000000
208	simpl exampl 2 thread separ thread 2 applic do...	You're on the right track exploring SslStream...	0.004764
248	risk honest lot safer handl anyth need encrypt...	The core issue lies in the use of .html() to ...	0.007681
72	host block think sql inject attack server need...	Identify the ModSecurity Rule:\nExamine the Mo...	0.008397
209	first read integ leav newlin charact input buf...	Solutions:\n\nConsume the Newline Character:\...	0.009264
101	said github3pi readm librari use hood specif e...	HTML Syntax Error (HTML1423):\nUse Browser De...	0.012155
275	correct syntax config.pi file pull valu use	Let's break down how to correctly pass pipeli...	0.012162

Key Observations:

- Similarity Range:AI and human answers are fundamentally different
- Extreme Cases:

  Human style: Straight to the point, no elaboration

  AI style: Educational, step-by-step guidance

Core Finding
Human answers: Efficiency-focused (sometimes too brief)
AI answers: Completeness-focused (always thorough)

reflection

Average semantic similarity is only ~0.14, with over 75% of AI–human pairs in low-similarity quadrants—high overlap is rare (<25%).

Low Phrasing Overlap

Consistent Positivity vs. Variability:

AI answers are slightly more positive (Δ≈0.03) and show tighter sentiment distributions, whereas human replies vary widely in tone and length.

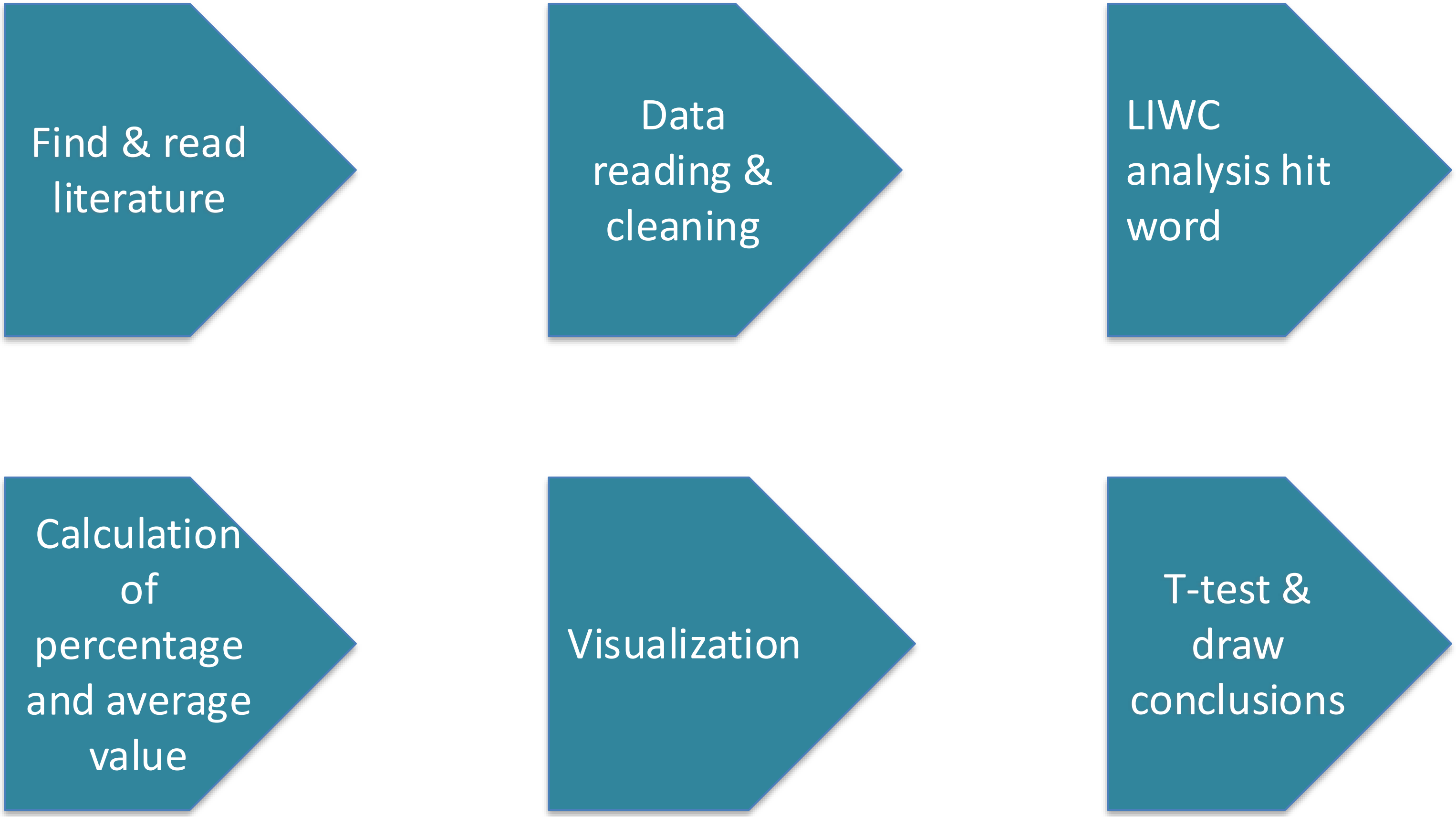
Depth & Style Contrast

Humans often drop in quick code snippets or direct answers; AI delivers uniform, step-by-step explanations using concise root-form vocabulary.

PART SIX

Using LIWC to Compare Linguistic Features of AI vs Human Answers

What I Did



Find & read literature

Data reading & cleaning

LIWC analysis hit word

Calculation of percentage and average value

Visualization

T-test & draw conclusions

Background and Motivation

- Humans and AI have different answering styles, and language style may affect learning outcomes
- This task aims to analyze the language differences between two platforms (Gemini vs Stack Overflow)
- Goal: Determine which answer is more systematic, rational, and suitable for knowledge transfer

- Use the LIWC tool to analyze three types of language features:
- Cognitive Processing, Positive Emotion, Negative Emotion
- Compare Gemini (AI) and Stack Overflow (human) in these three aspects

Methodology Overview

- Data reading & cleaning
- LIWC analysis hit word

- Dictionary of the LIWC:
- **Positive**["agree", "excite", "good", "great", "happy", "hope", "joy", "love", "ok", "positive", "pretty", "safe", "success", "yes", "like"]
- **Negative**[“afraid”, “angry”, “bad”, “confuse”, “cry”, “enemy”, “fail”, "fear", "frustrate", "grief", "hate", "kill", "nervous", "pain", "piss", "problem", "sad", "scare", "tense", "worry", "worthless"]
- **Cognitive**["always", "and", "because", "block", "but", "cause", "consider", "constrain", "could", "effect", "except", "explain", "guess", "hence", "if", "include", "know", "maybe", "never", "ought", "perhaps", "reason", "should", "think", "with", "without", "would"]

Proportion calculation

- Count the number of words belonging to each type of keyword in each answer text, calculate their proportion in the total vocabulary, and output the percentage.
- That is, the proportion of positive, negative and cognitive keywords matched in the answers of so and Gemini respectively in the total number of answer words.

The proportion of each type of words =
(the number of keywords matching a single answer / the total number of words in a single answer) %

Output example

answers_clean	Gemini_clean	answers_posemo_words	answers_negemo_words	answers_cogproc_words	gemini_posemo_words	gemini_negemo_words	gemini_cogproc_words
need encrypt hide communt…	the developer is concerned about transmitti…	[]	[]	['know', 'know', 'know’]	['like', 'like’],	[]	['and', 'but', 'if', 'and', 'and', 'should', 'and', 'and', 'and’]

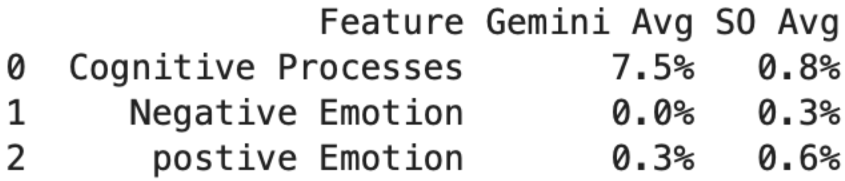
answers_word_count	gemini_word_count
88	144

answers_posemo	answers_negemo	answers_cogproc	gemini_posemo	gemini_negemo	gemini_cogproc
0.00%	0.00%	3.41%	1.39%	0.00%	6.25%

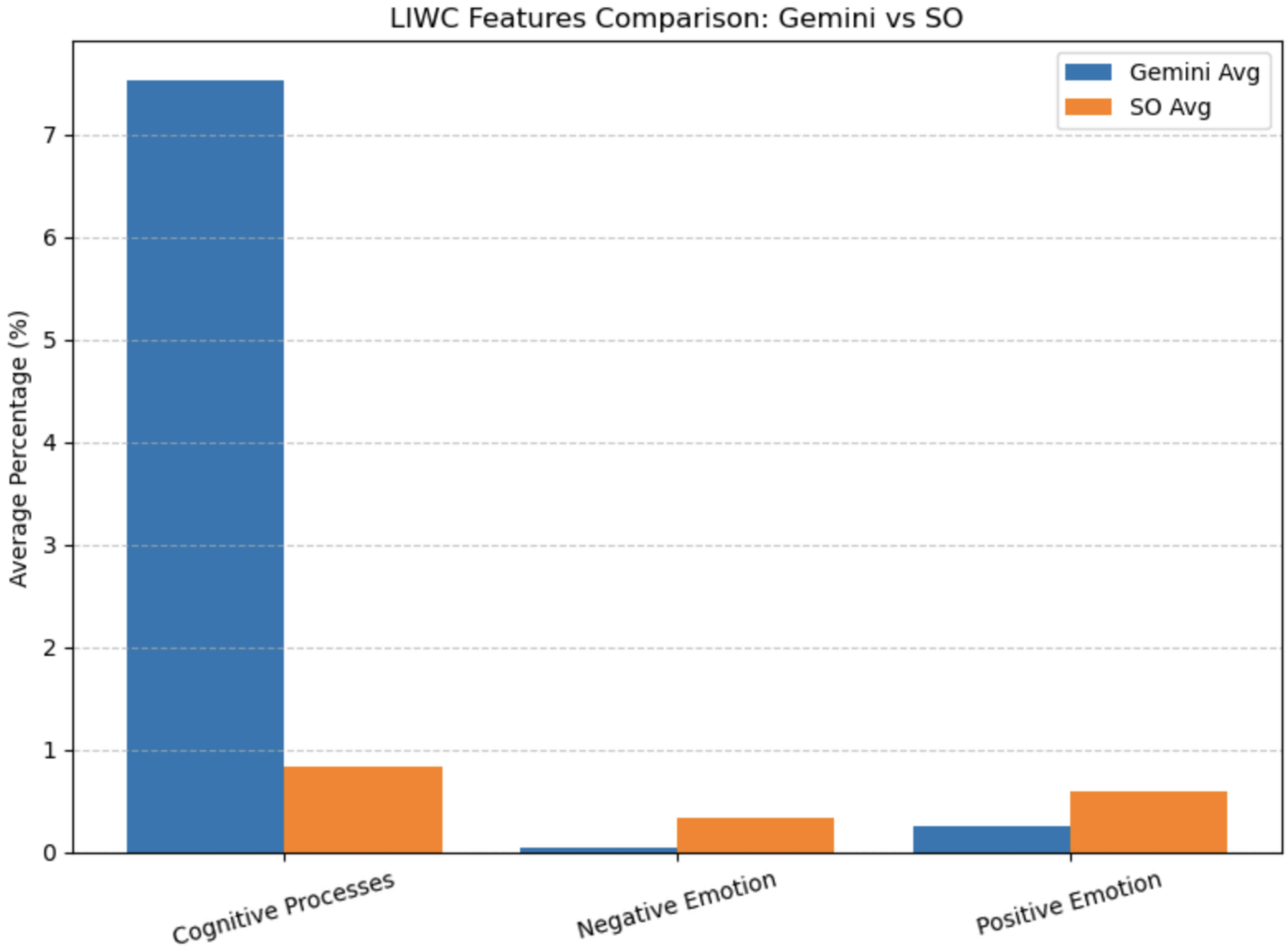
From file: liwc_analysis_with_words.csv

Average value calculation & visualization

Calculate the average of the proportion of each column of so and ai.

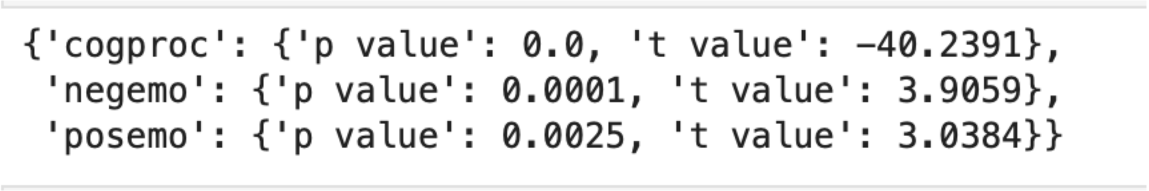


```
                  Feature Gemini Avg SO Avg
0   Cognitive Processes       7.5%   0.8%
1      Negative Emotion       0.0%   0.3%
2       postive Emotion       0.3%   0.6%
```



T-test

Is the Difference Significant?



```
{'cogproc': {'p value': 0.0, 't value': -40.2391},
 'negemo': {'p value': 0.0001, 't value': 3.9059},
 'posemo': {'p value': 0.0025, 't value': 3.0384}}
```

T value measures: the difference in the mean of the two groups / the overall degree of variation.
$t > 0$: the mean of the first group (OS) > the second group (Gemini)
$t < 0$: the mean of the first group (OS) < the second group (Gemini)

The **p-value** measures whether the difference you observed is likely to occur by chance.
$p < 0.001$ Highly significant difference
$0.001 \le p < 0.01$ Significant difference
$0.01 \le p < 0.05$ Marginal difference
$p \ge 0.05$ No significant difference
If $p < 0.05$, it indicates that there is a significant difference in this dimension.

The difference is not accidental and is statistically significant.

Conclusion and reflection

- Through language feature analysis, the differences in cognitive and emotional expressions between AI and human responses were revealed.
- The results show that AI is more rational and more suitable for knowledge transfer, supporting the overall goal of the project.

- The LIWC tool has a good reference value in academic analysis.
- In the future, it can combine more data and introduce texts from multiple languages and fields for more extensive analysis.

Thanks for listening