ISSN 1816-6075 (Print), 1818-0523 (Online)
Journal of System and Management Sciences
Vol. 14 (2024) No. 5, pp. 294-307
DOI:10.33168/JSMS.2024.0518

U-Tapis Web: An Automated Indonesian News Text Error Detection System

Rudi Sutomo[1], Marlinda Vasty Overbeek[2], Niknik Mediyawati[3], Samiaji Bintang Nusantara[3]

[1] Information System, Universitas Multimedia Nusantara, Tangerang, Indonesia

[2] Informatics, Universitas Multimedia Nusantara, Tangerang, Indonesia

[3]Digital Journalism, Universitas Multimedia Nusantara, Tangerang, Indonesia

rudi.sutomo@umn.ac.id (Corresponding Author), marlinda.vasty@umn.ac.id, niknik@umn.ac.id, samiaji.bintang@umn.ac.id

Abstract. This research developed U-Tapis Web, an automated Indonesian news text error detection system using the Flask framework. The tool implements linguistic algorithms for identifying spelling, syntactic, and conjunction errors. Evaluation across three studies with student samples demonstrated that the U-Tapis Web achieved high validity in detecting word use errors. The web interface enables user-friendly text analysis. Results showed participating students improved their language accuracy after utilizing the system. U-Tapis Web has promising applications for assisting news writing and language learning.

Keywords: news, detection words, framework flask, RUP

1. Introduction

In the field of reporting and news publishing, the accuracy and correctness of content is of paramount importance. News media strives to provide reliable and credible information to its readers. However, the challenge lies in the potential for word errors, which can weaken the quality and trustworthiness of news articles. One of the specific problems addressed by the U-tapis application is the prevalence of word errors in news content. These word errors cover many issues, including spelling errors, grammatical inaccuracies, and incorrect word choices. These errors can occur for various reasons, such as fast-paced news production, tight deadlines, or human oversight. While some word errors may be minor, others can significantly impact the overall clarity and credibility of a news article. This research is motivated by awareness of these challenges and the need for more specialized and accurate solutions. Additionally, the choice of the Flask framework for web programming and the MongoDB database for hosting further highlights the commitment to a user-friendly and scalable approach. This research also follows the Rational Unified Process (RUP) methodology, ensuring a structured and rigorous software development process. Integrating the Bootstrap framework simplifies the design process, making it more accessible to news editors and reporters. The U-Tapis application development combined previously created application designs with machine learning programs to process input words. The web design for U-Tapis was crafted using the Visual Studio application, and we also integrated the Bootstrap framework and the CSS programming language. Implementing the Bootstrap framework significantly expedited the application design process, as it eliminated the need to grapple with CSS syntax, resulting in responsive, lightweight, and straightforward applications (Yusuf Azhari, 2022).

U-Tapis web programming was carried out using the Flask framework, which was employed to merge the frontend application's appearance with machine learning programs written in the Python programming language. Additionally, for hosting this application, we opted for the MongoDB database. MongoDB is a popular and user-friendly database known for its flexibility, scalability, and simplicity, greatly facilitating the operation of algorithms/methods within the applications (Bonney et al., 2022).

The software development process in this research adhered to the Rational Unified Process (RUP) methodology in detail. This approach consisted of several phases of software development, including the inception, elaboration, construction, and transition phases. U-tapis seeks to bridge the gap in word error detection tools and methods for news content through deliberate choices and a blend of technologies. His specificity for the unique challenges of news writing and his commitment to accuracy make him valuable to journalism and language technology (Munir et al., 2016).

2. Literature Review

2.1. Sentence Syntax Error Detection U-Tapis Algorithm

Sentence Syntax Error Detection U-Tapis Algorithm is an algorithm that functions to detect Indonesian sentence syntax errors in a news article. This algorithm consists of a preprocessing algorithm, a Conditional Random Field (CRF) algorithm, and a Context Grammar (CFG) algorithm. The preprocessing algorithm processes news data into data per sentence that other algorithms can use. The Conditional Random Field (CRF) algorithm is used to assign tags/labels to each word/symbol in each sentence data. The Context Grammar (CFG) algorithm is used to parse each sentence's data and determine whether the syntax of a sentence is correct (if parsing succeeds) or incorrect (if parsing fails). This algorithm is run on a Virtual Private Server (VPS) owned by Multimedia Nusantara University (UMN) as an API (Application Programming Interface) using the Python Flask Web Framework. Web U-Tapis can call this API to check sentence syntax. CFG to outline the syntactic structure of the sentence and then use CRF for tagging words or entities in the context of the structure described by the CFG (Kornienko et al., 2021).

2.2. U-Tapis Algorithm for Detecting Conjunction Word Errors

The U-Tapis for detecting conjunction errors uses the Cosine Similarity algorithm to detect words that are conjunctions and determine whether the conjunctions are valid. Articles received will be pre-processed and tokenized into words to facilitate the detection stage. The Cosine Similarity algorithm calculates two vectors to get the similarity value between the two vectors. The words will be converted into vectors so that Cosine Similarity calculations can be applied to word tokens. The results of calculations using the Cosine Similarity algorithm determine whether the word token has a similarity value with the registered conjunctions. Validation for conjunctions is carried out based on the placement of the conjunctions consisting of prefix and intermediate conjunctions. The algorithm applied to U-Tapisfor detecting conjunction errors is implemented using Javascript with the Next.JS framework. The algorithm was also deployed using Vercel so that Web U-tapis could detect conjunctions (Mondi & Wijayanto, 2019).

2.3. U-Tapis Algorithm for Misspelling Word Detection

The algorithm used to detect spelling errors in the word teruah is Jaccard Similarity; this algorithm compares two pieces of data, which are then calculated using a formula called the Jaccard Index. The first data is in the form of articles received from the U- U-Tapis Web, which will then be pre-processed, such as changing all sentences to lowercase, removing excess space, removing punctuation marks, and tokenizing words from sentences. The second data is in the form of an Excel file that can be modified or added; this data will then be used as a dictionary containing all the correct luluh words (Tandiono, n.d.). Checking for spelling errors in the word through will be carried out if the following conditions are met:

1. Basic words must start with the prefixes s, p, t, and k.
2. The word will be checked if the word token length of the article is only 1 letter from the word in the dictionary.
3. The basic words of the word tokens from the article are the same as those in the dictionary.

If all the conditions above have been met, the Jaccard Similarity calculation will be carried out. The calculation result of Jaccard Similarity is 0 - 1; if 1, the word matches what is in the dictionary, then the word is appropriate or correct. This algorithm was implemented using Google Colab and then made into a microservice / API for use on the U-tapis Web.

One of the tools most often used for the website creation process is Visual Studio Code. Visual Studio Code is an application that creates, modifies, and deletes program code. Visual Studio Code is a Microsoft product that can currently be used for PC/Desktops with Windows, Linux and MacOS operating systems. Some of the advanced features that Visual Studio Code has include mastering various formats and programming languages, especially in the field of web development such as JavaScript, React, HTML, CSS, SCSS, JSON, Python, and others, IntelliSense which can complete code automatically, syntax highlighting, matching brackets, automatic indentation, even setting the type of font used to create code, the application background color, and the amount of spacing used.

3. Research Method

3.1. Fuse Inspection

During the inception phase, several key processes take place. These processes include business modeling and defining the problem's scope and boundaries. Additionally, the phase involves identifying system requirements to establish functional and non-functional software development needs. Furthermore, the inception phase includes an analysis and design stage, where an in-depth examination of workflow stages, data utilization, and previously determined functional and non-functional requirements is conducted. Lastly, the implementation process involves creating diagrams based on the established workflow and requirements (Munir et al., 2016) & (Christian & Erline, 2022).

3.2. Fase Elaboration

The elaboration phase, the second stage in software development using the Rational Unified Process (RUP), encompasses several vital activities. These activities include business modeling, data design, and user interface design. Business modeling involves a detailed exploration of business processes and requirements, resulting in comprehensive business models. These models define the project's scope, objectives, and boundaries, aligning the software with overall business goals and identifying key stakeholders and their needs. Data design is focused on structuring and organizing the data manipulated by the software, specifying data entities, relationships, and database architecture to ensure data integrity, efficiency, and scalability. Simultaneously, user interface design is pivotal in enhancing the user experience by creating an intuitive and visually appealing interface, considering layout, navigation, and user interactions to ensure user-friendliness and operational efficiency. Overall, the elaboration phase establishes a solid foundation, refining project scope and requirements and laying the groundwork for efficient data management and user-friendly interfaces, all contributing to the successful development of high-quality software (Munir et al., 2016) & (Ahmed & Ahmed, 2021).

3.3. Construction Phase

During the construction phase, several crucial processes occur, and one of the central activities within this phase is the actual construction of the program code. This involves implementing the code based on the previously designed architecture and specifications using the selected programming language. This step is where the software development team translates the conceptual design and functional requirements into tangible, executable code that will form the core of the software application. It requires meticulous attention to detail and adherence to coding standards to ensure the code is robust, maintainable, and functions correctly. Additionally, rigorous testing and quality assurance processes often accompany the construction phase to identify and rectify any bugs or issues that may arise during the coding process, ultimately leading to the development of a reliable and efficient software product (Sistem et al., 2023).

3.4. Transition Phase

The transition phase in software development encompasses several crucial processes to ensure the software's quality and readiness for deployment. These processes include business modeling, where the testing methodology is determined to address quality standards and potential issues. Additionally, the requirements are identified to procure the necessary equipment and resources, including tools and hardware, essential for the testing phase. Detailed test scenarios are designed based on use cases during this phase to provide comprehensive testing coverage. Finally, the software undergoes testing using these predefined use-case scenarios, allowing testers to evaluate its functionality, performance, and adherence to requirements. Any issues or defects discovered are documented and resolved, contributing to the software's overall readiness for successful deployment in real-world scenarios (Harmilasari & Munggaran, 2020).

3.4.1. Design and build a U-Tapis web system using the RUP method

U-Tapis application runs on the Ubuntu server. Ubuntu server does not require a lot of resources, has complete security features, and has a wide user community around the world, so this type is suitable for use in making web applications and databases that can be accessed online (Aulia et al., 2021) & (Ramadhan & Mukhaiyar, 2020).

3.5. Inception Phase

The inception phase is the first stage that needs to be carried out to identify the needs and features

needed in the software development process.

3.5.1. Business Modeling

The U-Tapis application employs a programming methodology with distinct functions in the process of language error detection, utilizing three algorithms: U-Tapis Syntax Sentence Error Detector, U-Tapis Conjunction Word Error Detector, and U-Tapis Syllable Spelling Error Detector. Each of these algorithms serves a specific purpose and collectively enhances the application's ability to identify and rectify various language errors (Mediyawati & Bintang, 2021).

3.6. Requirements

In order to fulfill the modeling of software, a comprehensive design is essential to meet the system's requirements, encompassing both functional and non-functional needs. Functional requirements represent the core components that must be addressed for a software system, which can be viewed in the table below:

Table 1. Functional Requirements

No.	Functional Requirements
1	The software can input several sentences as news
2	The software can check the sentences that are input with the three algorithms
3	The software can store the history of every incoming news and after checking.

Data requirements taken from Tribun News and one example of news that has been classified into a data table can be seen in the image below:

PENULISAN SALAH	PENULISAN BENAR	ALASAN	Nama/Entitas/Bahasa Asing	Keterangan	Tautan/Link
Diduga Ada Konflik Kepentingan Jelang Muktama	*Reshuffle*		Muktamar NU		tribunnews.com
Sambil	Ketika/Sembari		Presiden Jokowi		wartakotalive.com
me-reshuffle	me-reshuffle		*Reshuffle*		
Pasalnya	Pasalnya,		Menteri Agama		
reshuffle	*reshuffle*		Danang Triatmojo		
mem-booking	*mem-booking*		JAKARTA		
di reshuffle	*di-reshuffle*		Mahasiswa Nahdlatul Ulama (IMANU)		
Selain itu	Selain itu,		Istana Negara		
maupun	dan		Jakarta Pusat		
, maupun	walaupun/meskipun/biarpun		Selasa		
manapun	mana pun		Presiden Joko Widodo		
Bila	Apabila		Komisi Pemberantasan Korupsi (KPK)		
di reshuffle	*di-reshuffle*		Yaqut Cholil Qoumas		
Jika tidak di reshuffle maka kami	Jika tidak di-reshuffle , kami akan kerahkan ...		Yaqut		
Gita Sinaga Ogah Nikah Karena Beda Keyakinan,	Tolak Ajakan Kawin Habibi Hood		Kemenag		
namun	, tetapi beda keyakinan		Calon Ketua Umum PBNU		
beda keyakinan,	beda keyakinan membuat ...		Yahya Cholil Staquf		
namun	, tetapi akan tetap ...		Muktamar		
terhadap tuhan	terhadap Tuhan, dan ...		Pengurus Besar Nahdlatul Ulama (PBNU)		
diusianya	di usianya		Menag		
tapi			Koordinator Aksi IMANU		
tak			Rosi		
malah engga			IMANU		
Gamau tah			Lampung		
Jadi	Jadi,		Desember		
samau dipatokin,			Ketum PBNU		

Fig.1： News sorted by correct words and wrong words

3.7. Analysis and Design

The series of processes in the implementation phase is carried out by analyzing the system process, identifying the software's functionality and user needs and designing a flowchart diagram model.

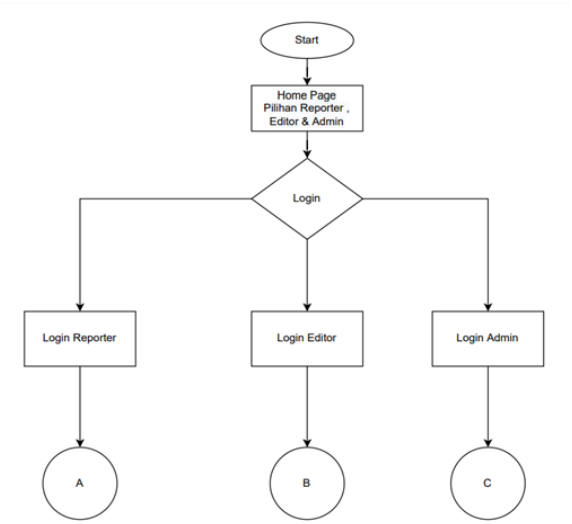

Fig.2: Design of the U-Tapis Web System

In the flowchart we examined, there is access to enter with permission as a Reporter, Editor, or Admin. Each role has features that have been adapted to each task. As a Reporter, you can enter news, while as an Editor, you can check the news that has been entered using the language error detection feature, and the Admin can monitor all processes.

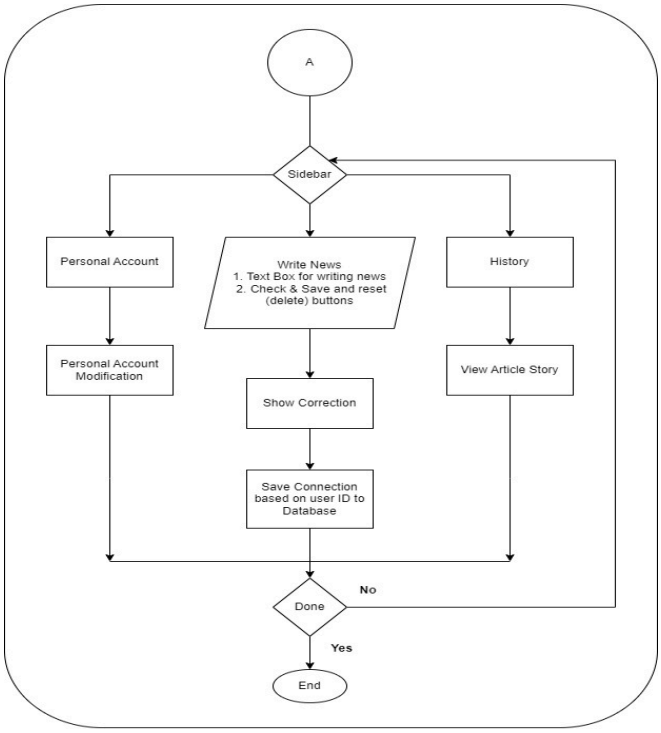

Fig.3: Design of U-Tapis Reporter Web System

In this Flowchart, reporters can enter news titles, news categories, and articles they have written to be checked by reporters using the U-filter system. After filling in the title, category, and news article, the reporter can press the "send" button in the screen's bottom right corner.

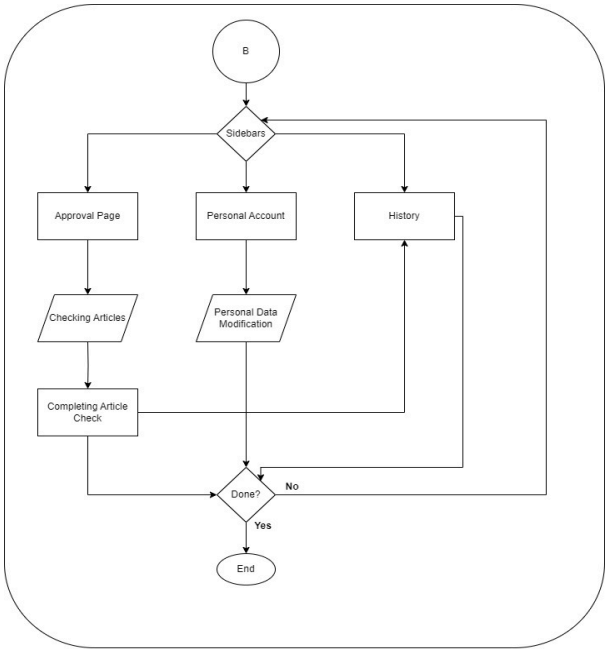

Fig.4: Design of U-Tapis Editor Web System

In this flowchart, editors can see what news reporters have submitted for review. To conduct a review, the editor only needs to click the "view" option on the news to be reviewed. Next, scroll down until you find the 'Delete' button. If the 'filter' button is pressed, the news will be processed using the U-filter system. Currently, U- U-Tapis can detect syntax errors in sentences, conjunction errors in sentences, and word derivation errors.

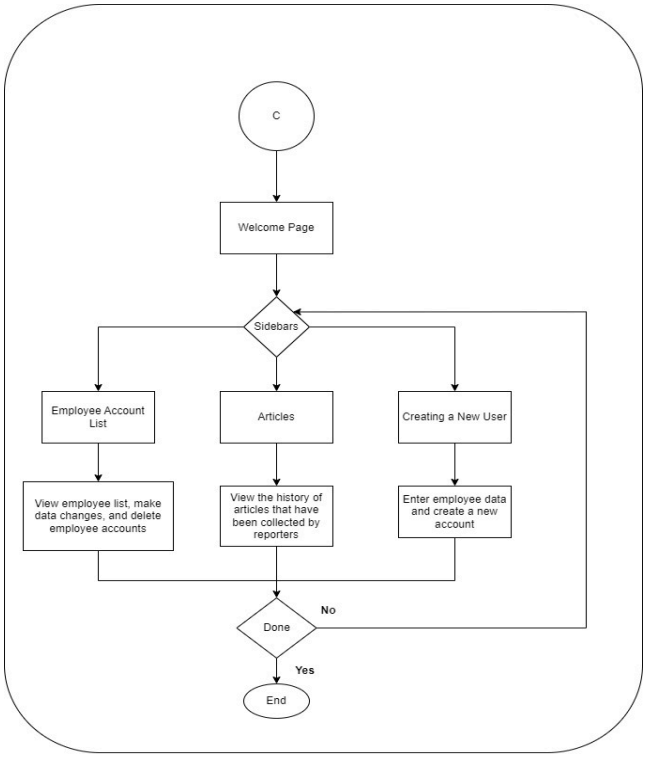

Fig.5: Design of Admin U-Tapis Web System

It contains a summary of the total news articles that have been collected, articles that the editor has approved, articles that are being edited, and a distribution diagram of news categories that have been approved.

3.8. Implementation

3.8.1. Elaboration Phase

This subchapter will show an overview of the software interface design that will be built. Figure 6 shows the main page display interface design in this study can be seen in the image below:

Fig.6: Display of the main page of the Admin U-Tapis web system

3.8.2. Construction Phase

This phase explains the programming language used and is detailed in the order below:

3.8.2.1. Framework Flask

(Ahmat Baseri et al., 2020)Flask is a web framework from the Python programming language. Flask itself has libraries and a collection of code that can be used to build websites without the need to do all the steps from scratch. This Flask framework is flexible and can support both front and back-end development. Making a website already uses the Flask Framework. The first step is to create HTML, CSS and JavaScript files with the help of the Bootstrap V framework. Bootstrap V is a framework that many front-end programmers use to make the website look more interactive and responsive for use by website users.

3.8.2.2. Using MonggoDB

The U-Tapis website itself uses the MongoDB Atlas database. MongoDB Atlas is a multi-cloud database service by the people who built MongoDB. MongoDB is a database system with better speed, flexibility, and ease of use than other database options. MongoDB on the website is a storage place for users and news data reporters to enter. Inside the MongoDB atlas consists of a cluster. Within the cluster is a collection, which is the database of the U-Tapis website. In total, there are 2 databases used. The first database is used to store user data, while the second database is used to store news. Before connecting from the U-Tapis website to the MongoDB Atlas database, you need the Pymongo module to use MongoDB in Python. The trick is to install Pymongo with the command "pip install Pymongo." after finishing installing Pymongo, we can call MongoClient with a URL or link from MongoDB Atlas, which can be taken from the MongoDB Atlas site itself so that the connection is successfully connected.

3.8.2.3. Connect the website to the Ubuntu VPS Server

(Mediyawati & Bintang, 2021)The U-Tapis website is deployed on UMN's Ubuntu VPS server. Website deployment starts with setting up flask website files into 1 package, which requires a file called MANIFEST.in, which contains the folders used in the U-Tapis website and a file called project. Tool, which contains the tools needed to build a package. After that, we can build by using Python –m build and creating a package containing the U-Tapis web.

The next step to connecting the U-Tapis website to the server is to download OpenVPN. After having OpenVPN, we can connect the network from the device to the UMN network. After connecting to the UMN network, the next step is downloading putty. After downloading Putty, we can

connect with the Ubuntu server from UMN and set up the IP address, namely 10.100.1.100 and log in with the username and password given, namely "utapis" in Putty.

4. Result and Discussion

In this phase, testing is carried out, and in the implementation of tests that have been carried out, you can see the test sequence below

4.1. First research (2020)

In the first study, U-Tapis was tested on 30 students from the Department of Journalism, Faculty of Communication Sciences, Multimedia Nusantara University in one batch in 2020. Prior to treatment with U-Tapis, respondents conducted an initial test with 7 indicators of language structure where errors were still frequently found, namely affixes, bound forms, compound words, prepositions, particles, standard words, and typos. Based on these 7 indicators, scoring is carried out from the results of the test given in the form of assignments to write journalistic works in the form of news and articles or opinions without giving any treatment to students. From these data, it can be seen that there are still many errors in the use of letters (14% capitals, italics 86%), word writing (38% affixes, 25% prepositions, 21% compound words, 10% bound forms, 4% particles, and numbers 2%), and the use of punctuation (commas 70%, periods 23%, and hyphens 7%). In addition, errors in using effective sentences also occur (55% inaccuracy, 37% inaccuracy, and 8% disharmony).

The design of the U-Tapis application is only the first stage. Hence, the filter's capacity is still limited to writing words consisting of affixes, bound forms, compound words, prepositions, particles, standard words, and typos. Therefore, special examples of word writing errors will be shown that are repeated by UMN Journalism students.

Table 2. List of Inaccurate Use of Indonesian

No.	Words Wrong Affix	Bound Form	Compound Words	Front Particle Words	Baku words
1.	menggerakan	maha kuasa	terimakasih	Kedalam demikian lah	karir
	olahlaga				
	(menggerakkan)	(mahakuasa)	(terima kasih)	(ke dalam) (demikianah)	(karier)
	(olahraga)				
2.	di panggil	non teknis	kerjasama	Diluar diapun	dipungkiri
	anilah				
	(dipanggi)	(nonteknis)	(kerja sama)	(di luar) (dia pun)	(dimungkiri)
	(inilah)				
3.	menunjukan	pra produksi	tanggungjawab	Diatas masih kah	didisain
	(menunjukkan)	(praproduksi)	(tanggung jawab)	(di atas) (masihkah)	(didesain)
4.	di jadikan	pasca produksi	sepakbola	Disini meski pun	cidera
	name				
	(dijadikan)	(pascasarjana)	(sepak bola)	(di sini) (meskipun)	(cedera)
	(namun)				
5.	mensubsidi	antar kota	tindaklanjut	Dirumah apapun	respon
	karna	(menyubsidi)	(antarkota)	(tindaklanjut) (di rumah)	(apa
	(response)	(karena)			

6.	merubah	pasca nikah	beritahu	Dibidang malampun	manager
	masuh				
	(mengubah)	(pascanikah)	(beri taahu)	(di bidang) (malam pun)	(manajer)
	(masih)				
7.	pertunjukkan	anti sosial	orangtua	dibawah apa lah	professional
	uluh				
	(pertunjukan)	(antisosial)	(orang tua)	(di bawah) (apalah)	(profesional)
	(utuh)				
8.	berpergian	non muslim	bulutangkis	diantara dia lah	nafas
	melajutkn				
	(bepergian)	(nonmuslim)	(bulu tangkis)	(di antara) (dialah)	(napas)
	(melanjutkan)				

On that basis, it is necessary to treat the use of the U-tapis application to minimize these errors. To measure students' abilities, a final test was carried out with the help of the U-tapis application. Based on 30 previously collected journalistic papers, the system found 130 of the 137 errors made (around 95% of errors were made). Furthermore, based on 130 errors made, the system provided correct corrections 87 times (approximately 66.92% correct corrections of all detected errors). Even though the system cannot provide correct correction suggestions for 33.08% of errors made by students, based on the system's highlighting process, students are indirectly given the opportunity to double-check the KBBI and PUEBI to correct the mistakes they made.

Based on further investigation, errors that failed to be corrected by the U-tapis system consisted of (1) typing errors in the foreign words used which were not italicized, (2) typing errors containing missing spaces between two words, (3) typing errors in named entities that do not begin with a capital letter. On the other hand, errors that the U- U-Tapis system cannot find consist of words that are not commonly used and therefore are not included in the reference dictionary used by the system and typing errors in a word starting with a capital letter and typing errors in a word starting with a capital letter (because U- Tapis assumes that words starting with a capital letter are an entity, the system does not carry out the correction process).

Based on the data from the final test results, it can be seen that the level of students' word errors has decreased in terms of the achievement indicators and scores obtained. Furthermore, it can also be seen that there is an average difference between before and after treatment with U-Fix. Therefore, the next step is to compare the pre-test and post-test scores. Based on the test requirements for data analysis, namely statistical tests using the normality test and homogeneity test using the SPSS 25 application, the significance was $0.111 > 0.05$ and $0.176 > 0.05$, so the distribution of the data before and after treatment was normally distributed, whereas based on the homogeneous test, a significance value of $0.143 > 0.05$ means that the two data are homogeneous.

This research uses a one-group pretest-posttest design experiment. This means that this research aims to see whether there is a difference in the average of two paired and related samples. Next, a hypothesis test is carried out using the Paired sample t-test with the help of SPSS 25, and the results show that the average score of the mean of students before treatment was 24.97, and after being given treatment, it was 53.00. This means there is a significant increase in the average score after treatment using U-Tapis. It can be determined that the correlation is 0.076, so it can be said that the relationship

between the increase before and after being given treatment is quite close. Based on the level of effectiveness, there is a significance of 0.000 <0.05, so there is a significant difference between before and after administration of U-Fix. This means that the student error rate decreases, and the success of U-Tapis as an application that helps in writing can be effective.

Table 3. Comparison Results of Preliminary and Final Tests

Paired Samples Test

		Paired Differences					t	df	Sig. (2-tailed)
		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference Lower	Upper			
Pair 1	Sebelum - Sesudah	-28.03333	4.33497	.79145	-29.65204	-26.41463	-35.420	29	.000

4.2. The Second Research (2021)

Based on the results of the second study, which used 82 news articles processed using U-tapis, ignoring foreign words and entity names (because the current system does not cover them), 62 articles contained errors. They could be detected by the system as follows.

Table 4. Ability of U-Tapis in Detecting Language Errors

NO.	TEMUAN U-TAPIS	KET
1.	corona ==> korona | efikasi ==> fiksasi | efikasi ==> fiksasi |	
2.	dicalokan ==> dicalonkan | bukun ==> bukan	
3.	idol ==> idola | nggak ==> enggak | tuh ==> itu | itunya ==> tuanya | boy ==> bom | gitu ==> situ | idol ==> idola | idol ==> idola | gimana ==> dimana | nggak ==> enggak | fans ==> anus | idol ==> idola | udah ==> sudah | gimana ==> dimana | nggak ==> enggak | kesenengan ==> kesenangan | for ==> ori |	
4.	teehadap ==> terhadap | komika ==> kosmik | temen ==> teman | tapi ==> tadi | tapi ==> tadi | nemu ==> temu | dress ==> reses |	

Based on the U- U-Tapis findings table, the detected words and their improvements can be summarized as follows. Efficacy becomes fixation (2 times), book becomes not (1), idol becomes idol (4), egg becomes no (4), tuh becomes that (6) that becomes old (1), that becomes there (1), how becomes where (2), already becomes already (5), fun becomes pleasure (1), facing becomes towards (1), comic becomes cosmic (1), friend to be friends (1), but to be (11), meet to be meeting (1), use to be a package (1), financial to be financial (1), pandemic to be pandemic (1), like that to be that (2), not to be no (8), comment becomes moment (1), choose becomes own (1), followed becomes followed (1), lease becomes asean (1), no becomes no (8), find becomes discovery (1), win becomes cry (1), disappointment becomes disappointment (1), repeat becomes repetition (1), praise becomes word (1), awarded becomes awarded (1), national becomes national (1), motivate becomes innovate (1), promo becomes chromo (3), disorder becomes disorder (1), very becomes very (1), extra becomes extra (1), curricular becomes curricular (1), resistant becomes resistant (1), explains explains (1), ensures becomes ensures (1) , his widow became his sign (1).

Furthermore, U- Tapis can also detect and correct words such as imam becomes mama (1), night becomes night (1), selective becomes selective (1), busy becomes busy (1), career becomes cashier (1), ruin becomes destruction (3), career becomes career (2), pray becomes domain (1) namely becoming namely (1), audition becoming addiction(1), case becoming litigating (1), no becomes no (1),

administration becomes administration (1), abalabal becomes random (1), signal becomes signal (1), billion becomes billion (1), no becomes unwilling (1), lost becomes advertisement (1), thinking becomes thinking (3), not being me (1), saving becomes saving (1), external becomes born (1), pleading pleading (3), prokes becomes process (3), ambyar becomes image (1), consumption becomes consumption (1), until it becomes a sample (3), want becomes want (1), hurry becomes hurry (1), conversation becomes conversation (1), no becomes it (3), expectation becomes expectation (1), archive becomes archive (1), integrated becomes integrated (1), application becomes application (1), application becomes application (1), ministry becomes ministry (1), daan becomes will (1), to say becomes invite (1), string becomes barrage (1), pautin becomes putting (1), jg becomes ji (1), sidah becomes already (1), and untul becomes for (1).

Then, U- Tapis can also detect and correct words such as watch becomes watch (4), the queue becomes queue (3), thru becomes know (3), the restaurant becomes resort (1), sempet becomes time (1), not yet become yet (1), systemic (1), breathing becomes breathing (1), acting becomes so (2), the audience becomes audience (1), karuan becomes chaotic (1), they become clean (1), obey becomes Mentawai (3), synergy becomes energy (1), energy becomes energy (1), socialization becomes socialization (1), pre-construction becomes apaconstruction (1), prestigious becomes prestigious (1), waist becomes grilled (1), national team becomes team (1), he becomes displayed (1), what he does becomes achieved (1), noodles become ideas (1), chatting becomes chatting (1), tape becomes a type (3), claims become claims (1), kepdaa becomes to (1), circumventing becomes circumventing (1), and licensing becomes licensing (1).

It has also been detected that words such as in the realm become arrows (1), think becomes thought (1), single becomes singlet (3), first becomes a helping hand (1), but becomes a cow (4), the sub-subjects become the issues (1), sub becomes issues (1), so much becomes so (1), neat becomes fragile (1), becomes (1), paradikma becomes paradigm (1), which becomes yang (1), millennial becomes millennial (1), the question becomes personal (1), Millar becomes billion (1), milliard becomes millibar (1), must become must (1), also becomes (1), the response becomes response (1), visible becomes visible (1), fake becomes fake (1), and self becomes released (1).

Based on 82 news articles that were processed using U-Tapis, ignoring foreign words and entity names (not being the scope of the current system), there were 216 errors detected by the system. Of the 216 errors found, 39 errors were invalid (words are standard words found in KBBI), and 177 other errors were valid. Through this ratio, it can be concluded that U- U-Tapis has a detection validity value of 81.94%. Then, further analysis showed that of the 177 valid errors detected, U-Tapis succeeded in making suggestions for improvements to 93 words.

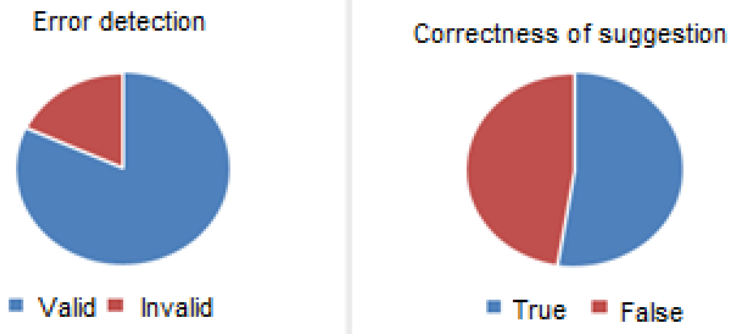

Fig.7: Errors and Corrections Detected by U-tapis

On the other hand, to find the 216 errors, the U-Tapis system only takes 0.709 seconds or around 0.0086 seconds per article.

The effectiveness of U-Tapis is based on 7 indicators of linguistic structure where mistakes are still often found, namely affixes, bound forms, compound words, prepositions, particles, standard words, and typos. Apart from that, the U-tapis character is also seen as follows.

1) U-Tapis can detect and correct typo errors (Celtic), such as the ones that also become the release, and socialization becomes the socialization.
2) U-Tapis can detect and improve the process of forming affixed words, such as tricking into tricking, busying to busy, and ensuring to be sure.
3) U-Tapis can detect and correct standard words into non-standard such as response to response, breath to breath, chatting to chatting, queue to queue, career to career, and permit to permit.
4) U-tapis has not been able to detect non-formal language or slang used in Tribunnews.com articles. Especially the -in the suffixes, such as what becomes achieved, lie in becomes a lie, find becomes to find, and win becomes a cry.

4.3. Third Research of U-Tapis Web (2022)

In the third research, 27 students majoring in Journalism, Information Systems and Information Engineering tested the U- U-Tapis Web, U- U-Tapis. In this third research, Web U-tapis, a website-based application, accommodates a program that can carry out word corrections mentioned in the previous research.

The development of the U-tapis application was built by combining the application design created with a machine learning program to process the input words. Web design was created using the Visual Studio application, and we also implemented the Bootstrap framework with the CSS programming language. Implementing the Bootstrap framework makes application design work faster because you no longer need to think about CSS syntax and produce a responsive, light and simple application.

In this third research, the Website team was tasked with creating a fast and responsive application. In terms of programming, the U- U-Tapis Web is carried out using the Flask framework, which combines the application display from the front end with machine learning programs that use the Python programming language. On the other hand, the database used to accommodate this application is the MongoDB database. MongoDB is a database that is quite popular and easier to use. MongoDB Database has flexibility, scalability and simplicity, making it easier for users to operate their applications.

The U-tapis application is run on an Ubuntu server. Ubuntu server does not require a lot of resources, has complete security features, and has a wide user community worldwide, so this type is suitable for creating websites, applications, and databases that can be accessed online.

5. Conclusion

Thus, it can be concluded that U- U-Tapis has the potential to improve language writing skills and detect errors in writing. The first research shows that using U-Tapis can significantly increase students' scores in terms of language structure. The second study indicated that U-tapis had a high level of detection validity, especially in detecting word errors in news texts. The third research shows further development with web-based application development. This study makes important theoretical and practical contributions regarding

automated linguistic error detection systems. The successful development of Utapis Web provides an effective text analytics tool for identifying common Indonesian language errors. Testing demonstrated its capabilities for enhancing writing accuracy across diverse error types. The web-based accessibility promotes widespread adoption. However, limitations exist in the scope of the algorithms and dataset. Future work should expand the system's capabilities to detect more complex lexical, grammatical, and semantic issues. Incorporating larger news corpora for training and testing would also boost performance. Longitudinal classroom studies are needed to assess long-term learning outcomes. Overall, the research enables significant progress in automated linguistic analysis, with myriad applications from journalism to education.

Acknowledgment

The authors would like to thank Universitas Multimedia Nusantara for the support given to this research.

References

Ahmat Baseri, N. B., Bakar, J. A., Ahmad, A., Jafferi, H., & Zamri, M. F. (2020). SMVS: A Web-based Application for Graphical Visualization of Malay Text Corpus. *ISCAIE 2020 - IEEE 10th Symposium on Computer Applications and Industrial Electronics*, 30–35. https://doi.org/10.1109/ISCAIE47305.2020.9108705

Ahmed, J., & Ahmed, M. (2021). Online News Classification Using Machine Learning Techniques. *IIUM Engineering Journal*, *22*(2), 210–225. https://doi.org/10.31436/iiumej.v22i2.1662

Aulia, T. M. P., Jamaludin, A., & ... (2021). Extractive Text Summerization Pada Berita Berbahasa Indonesia Menggunakan Algoritma Support Vector Machine. *J-SAKTI (Jurnal Sains …*, *5*(September), 727–735. http://ejurnal.tunasbangsa.ac.id/index.php/jsakti/article/view/371

Bonney, M. S., De Angelis, M., Dal Borgo, M., Andrade, L., Beregi, S., Jamia, N., & Wagg, D. J. (2022). Development of a digital twin operational platform using Python Flask. *Data-Centric Engineering*, *3*(1), 1–14. https://doi.org/10.1017/dce.2022.1

Christian, Y., & Erline, M. (2022). Web-Based Chatbot With Natural Language Processing and Knuth-Morris-Pratt (Case Study: Universitas Internasional Batam). *JST (Jurnal Sains Dan Teknologi)*, *11*(1), 132–141. https://doi.org/10.23887/jstundiksha.v11i1.43258

Harmilasari, D., & Munggaran, C. (2020). Evaluasi Kepuasan Pengguna Portal Berita Menggunakan Usability Metric. *Jurnal Ilmiah Komputasi*, *19*(3), 293–300. https://doi.org/10.32409/jikstik.19.3.23

Kornienko, D. V., Mishina, S. V., Shcherbatykh, S. V., & Melnikov, M. O. (2021). Principles of securing RESTful API web services developed with Python frameworks. *Journal of Physics: Conference Series*, *2094*(3). https://doi.org/10.1088/1742-6596/2094/3/032016

Mediyawati, N., & Bintang, S. (2021). Platform Kecerdasan Buatan Sebagai Media Inovatif Untuk Meningkatkan Keterampilan Berkomunikasi: U-Tapis. *Prosiding Seminar Nasional Pendidikan Program Pascasarjana Universitas PGRI Palembang 21 Augustus 2021*, 69–79.

Mondi, R. H., & Wijayanto, A. (2019). Recommendation System With Content-Based Filtering Method for Culinary Tourism in Mangan Application. *ITSMART: Jurnal Ilmiah Teknologi Dan Informasi*, *8*(2), 65–72.

Munir, M., Mahali, M. I., Dewanto, S. A., Wulandari, B., & Hasanah, N. (2016). Pengembangan Smart Traffic Light berbasis IoT dengan Mobile Backend as a Service sebagai wujud Smart City bidang transportasi. *Elektronika Dan Informatika*, 1–15.

Ramadhan, R. F., & Mukhaiyar, R. (2020). Penggunaan Database Mysql dengan Interface PhpMyAdmin sebagai Pengontrolan Smarthome Berbasis Raspberry Pi. *JTEIN: Jurnal Teknik Elektro Indonesia*, *1*(2), 129–134. https://doi.org/10.24036/jtein.v1i2.55

Sistem, P., Penilaian, I., Universitas, M., Ulama, N., Barat, S., & Web, B. (2023). *JURNAL TEFSIN*. *1*(1), 7–12.

Tandiono, J. (n.d.). *Detection of Melt Word in Indonesian Language Online News Portal Using The Rabin-Karp Algorithm (Case Study : Tribunnews)*.

Yusuf Azhari, R. (2022). Web Service Framework : Flask Dan Fastapi. *Technology and Informatics Insight Journal*, *1*(1), 58–65. https://doi.org/10.32639/tiij.v1i1.54