Faktz

INTI International University

FINAL YEAR PROJECT REPORT

"A dissertation submitted in partial fulfillment of the requirement of an

Faktz: An NLP and Machine Learning-Based Platform for Content Credibility on TikTok

Honors Degree in Information Technology at
INTI International University under the management and supervision of
Faculty of Data Science and Information Technology"

I declare that this project is my own work, it has not been copied in part or in whole from any source except where duly acknowledged. As such, all uses of previously published works (from books, journals, internet, etc.) have been properly acknowledged within the report to an item in the references or bibliographies. I hereby submit my dissertation, dated 11 / 23 / 2025 for review and assessment.

by
Student Name : Mayar Abu Latifa
Student ID : I25036233
IC No/ Passport No : U0171595
Project ID : FDSIT-INTI-IU-BITI-AUG-2025-0037

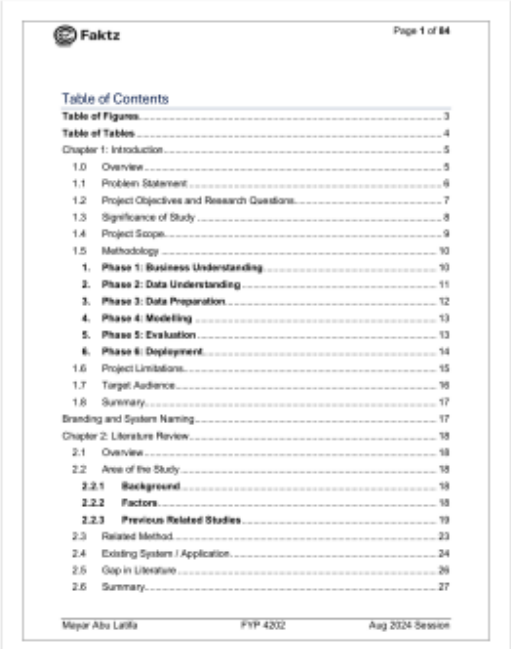
turnitin

Digital Receipt

This receipt acknowledges that Turnitin received your paper. Below you will find the receipt information regarding your submission.

The first page of your submissions is displayed below.

Submission author:	MAYAR ATALLAH IBRAHIM ABU LATIFA
Assignment title:	Final Year Project Documentation (Turnitin) - AUG2025
Submission title:	I25036233_MayarAbulatifa_turnitin.docx
File name:	I25036233_MayarAbulatifa_turnitin.docx
File size:	11.19M
Page count:	84
Word count:	20,641
Character count:	125,369
Submission date:	23-Nov-2025 10:56PM (UTC+0800)
Submission ID:	2824798393



Faktz Page 1 of 84

Table of Contents

Mayar Abu Latifa FYP 4202 Aug 2024 Session

Table of Contents

Table of Figures

Table of Tables

Chapter 1: Introduction

1.0 Overview

Social media platforms have become our second reality, a virtual world where people can express themselves freely, connect across continents with a single click, and build communities that transcend physical boundaries. For many, these platforms have created opportunities for jobs, income, friendships, and even lifelong partnerships. The beauty of social media lies in this freedom of expression and discussion, allowing individuals to be themselves anytime and anywhere.

Yet the very freedom that defines social media also introduces significant challenges. A growing number of users now treat online content as a primary source of information, whether they are seeking health advice, lifestyle tips, political updates, or general knowledge. This shift creates a difficult environment where reliable insights are mixed with inaccurate claims, personal beliefs, and unverified statements. Many creators present information confidently even when it lacks scientific or factual grounding, and viewers often accept it without question due to the fast and immersive design of these platforms. As a result, distinguishing credible information from misleading content has become increasingly complex.

TikTok illustrates this tension clearly. It has transformed social media through short, engaging videos that reach wide audiences within minutes. Its trends, sounds, and creative tools give users the power to influence discussions, shape opinions, and form online communities at a rapid pace. However, this speed also allows misinformation to spread before users have the chance to verify what they are watching. Videos related to health, politics, finance, or personal development can easily give the impression of authority, even when the information is taken out of context or based on unreliable sources.

This project proposes a structured response to these challenges through the development of a credibility assessment platform known as Faktz. The system combines Natural Language Processing techniques and Machine Learning with modern web technologies to analyze TikTok video transcripts and classify them as opinions, claims, or supported claims. The platform is designed not only to display AI generated results, but also to provide a space where users can discuss credibility assessments, exchange ideas, and contribute to a shared understanding of online content.

1.1 Problem Statement

TikTok's recommendation system is designed to maximize engagement, which means that videos generating strong emotional reactions tend to reach wider audiences regardless of their factual accuracy. This creates an environment where opinions presented with confidence or claims lacking reliable evidence can easily appear credible to viewers. The short and fast paced nature of the platform makes verification difficult, as users often move from one video to the next without pausing to question the information they encounter. Studies have shown that many TikTok videos offering advice, factual statements, or interpretations of current events feature misleading content or references that cannot be traced back to trustworthy sources [1].

This challenge extends beyond individual users. Content creators rarely receive meaningful indicators regarding the credibility of what they share, which limits their ability to produce reliable and responsible content. Brands and marketing agencies that rely on TikTok for influencer partnerships face reputational risks when creators unintentionally spread inaccurate or misleading information. The consequences of misinformation are not limited to digital misunderstandings. In recent years, TikTok has even faced restrictions or bans in several countries due to concerns about harmful content, privacy risks, and its growing influence over public opinion. These actions reflect global unease about the role the platform plays in shaping beliefs and behaviors, particularly when credibility cannot be easily assessed [2].

Current Artificial Intelligence (AI) based approaches to misinformation often attempt to identify false information directly, yet they rarely distinguish between personal opinion, claims, and evidence-based statements. This limitation reduces their usefulness in a platform as dynamic and context dependent as TikTok. A more nuanced and accessible system is needed to help users interpret content responsibly without limiting the diversity of voices that contribute to the platform's appeal.

In summary, the problem lies in the absence of a practical and transparent tool that helps audiences understand the credibility of TikTok content. The platform's fast paced design allows unverified material to spread rapidly, leaving users, creators,

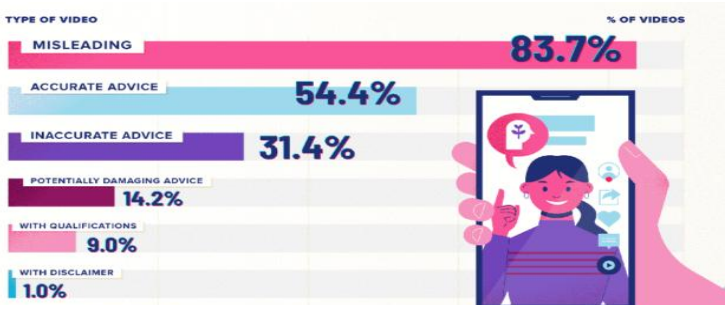


Figure 1: Percentage of misleading and inaccurate mental health advice on TikTok videos [3]

and brands without reliable guidance to assess the trustworthiness of what they encounter.

1.2 Project Objectives and Research Questions

Table 1: Project Objectives Vs Research Quetsion

Project Objective	Research Question(s)
To identify and define appropriate credibility labels that effectively differentiate TikTok content based on evidence, linguistic cues, and communication style.	Can conceptual and linguistic criteria distinguish speech in short form videos?
To collect, clean, and annotate TikTok transcripts to build a reliable, high-quality dataset for credibility classification.	1. Which data sources and collection methods best capture the language style found in real TikTok videos? 2. What cleaning and preprocessing techniques ensure consistency while preserving the natural structure of TikTok style text? 3. How can annotation guidelines be designed to clearly separate labels in a manner that mirrors actual content found on the platform?
To develop and train an NLP based ML model to classify TikTok transcripts into credibility categories.	1. Which NLP and ML model best suit this classification task? 2. What linguistic features improve the model's ability to separate the three credibility categories?
To evaluate model performance and system usability through validation metrics and user feedback.	Which evaluation metrics best reflect real world model effectiveness?
To deploy the trained model into a web-based platform.	How can credibility classifications be displayed in an interpretable and user-friendly format?

These objectives guide the project toward building a clear and reliable way to understand credibility on TikTok. They begin by defining the labels needed to separate different types of content, followed by creating a dataset that reflects how people naturally communicate on the platform. The project then explores suitable NLP and machine learning techniques to identify the model that performs best for classification.

Once developed, the model is integrated into an interactive web platform that presents results clearly to users. The final stage evaluates the system through technical metrics and user feedback to ensure it is accurate, practical, and useful in real settings.

1.3 Significance of Study

TikTok has grown into one of the most powerful social platforms worldwide. With more than one billion active users and rising engagement levels, it has become a central space where people spend a significant part of their day. Recent reports show that the average user spends nearly ninety-five minutes daily on TikTok, often returning to the app many times throughout the day. These habits reveal how deeply the platform has become woven into everyday life, especially among younger generations, figure (2) below shows some TikTok statistics that highlight its influence.

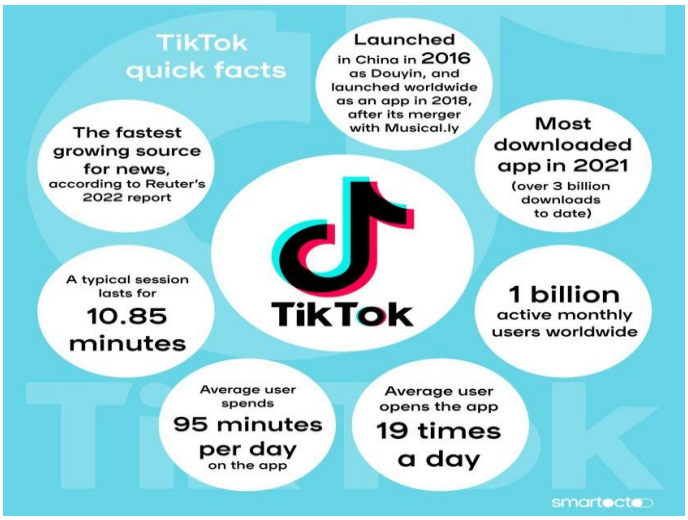


Figure 2: TikTok quick facts highlighting its global growth and user engagement [3]

For Generation Z, TikTok represents far more than entertainment. It is a place where ideas are shared openly, creativity is celebrated, and new forms of income and influence are created. Many young individuals use TikTok to build careers, connect with global communities, and express themselves freely. The platform has also quietly become an informal learning space where users encounter health tips, political perspectives, lifestyle advice, and explanations of current events. This openness has allowed information to spread faster than traditional channels, making TikTok an important part of how many people understand the world.

However, this speed also creates challenges. Content can go viral long before its accuracy is questioned, and unverified claims often circulate as if they were facts. These concerns have grown so large that several countries have placed restrictions or bans on TikTok due to worries about misinformation and the platform's influence on public opinion. The situation highlights a global need for better ways to understand the credibility of the content that spreads so rapidly online.

This study is significant because it provides a practical response to this growing issue. By establishing clear credibility labels, preparing a dataset that reflects how people actually communicate on TikTok, and building a model that categorizes content into meaningful groups, the project creates a structured way to help users interpret what they see. The aim is not to limit creativity or silence voices, but to offer a layer of clarity that supports healthier and more responsible engagement. Through this approach, the system helps users, creators, and organizations navigate an environment where information moves quickly and verification is often overlooked.

1.4 Project Scope

The scope of this project is centered on the development of a lightweight credibility classification model that operates on TikTok video transcripts. The work begins with collecting raw transcript data from reliable and diverse sources that reflect real TikTok communication. This includes merging datasets, removing inconsistencies, and applying preprocessing steps that produce clean and structured text suitable for machine learning tasks.

A key component of the scope involves designing and preparing the complete data pipeline. This includes cleaning the transcripts using text normalization and pattern correction techniques, annotating them according to defined credibility labels, and ensuring that the final dataset is representative of typical TikTok content. The project also focuses on selecting and evaluating suitable Natural Language Processing (NLP) and Machine Learning (ML) techniques to determine the most effective approach for credibility classification.

The outcome of this phase is the creation of a lightweight NLP model that can distinguish between opinions, claims, and supported claims. The model is then optimized and prepared for integration into the web-based platform developed for this system. The scope concludes once the model is ready for deployment and can be accessed by the website through an appropriate API.

The project does not extend to frontend design, user interface development, or broader platform functionalities. It also does not include audio or visual analysis of TikTok videos. The primary aim is to deliver a clean dataset and an efficient, deployable NLP model that forms the foundation of the credibility assessment system.

1.5 Methodology

This project follows the CRISP-DM framework, which offers a practical and structured way to develop data-driven systems. The model supports a clear workflow from understanding the problem to preparing the final classifier for deployment. Each phase contributes to building a credibility classification component that is both reliable and suitable for real-world use, especially in the fast and dynamic environment of TikTok [4].

The CRISP-DM model consists of six major phases:

- Business Understanding
- Data Understanding
- Data Preparation
- Modeling
- Evaluation
- Deployment

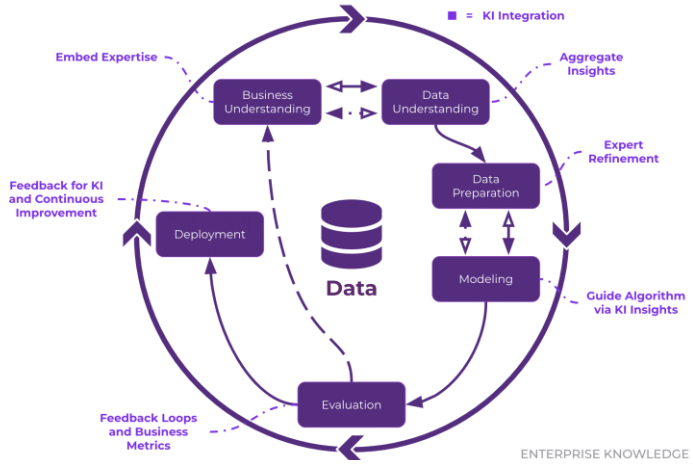


Figure 3: CRISP-DM Phases [4]

1. **Phase 1: Business Understanding**

   The first phase focuses on understanding why credibility assessment on TikTok has become an important research problem and how a classification system can support users, creators, and organizations. TikTok's engagement driven design allows attention grabbing videos to spread rapidly, which often results in opinions and unsupported claims being consumed as if they were verified facts. This creates challenges for viewers who depend on the platform for information and for brands that rely on influencers whose content may shape public perception.

   This phase adopts a mixed methods approach, combining qualitative exploration with quantitative preparation. The qualitative component involves examining how information is communicated on TikTok, observing patterns across different content categories, and identifying linguistic cues that signal whether a statement reflects personal opinion, an unsupported claim, or a reference to credible evidence. This exploratory analysis helps illuminate how users present information and why certain content types gain traction.

Alongside this, the study incorporates quantitative elements, drawing on structured datasets and measurable features that can later be used for machine learning classification. Together, these approaches define the project's central guiding question, which asks whether a lightweight NLP model can meaningfully separate TikTok transcripts into distinct credibility levels.

To support this foundation, existing academic research on misinformation, claim detection, influencer credibility, and social media linguistics were reviewed. These insights helped identify the features that shape the credibility labels used throughout the project.

By the end of this phase, the project's purpose, scope, and expected outcomes were clearly established, ensuring that the work remains rooted in real world needs while contributing to a more transparent and responsible digital environment.

2. Phase 2: Data Understanding

The second phase will focus on building a clear understanding of the transcript data that will form the foundation of the credibility classification system. Since TikTok communication is informal, fast paced, and highly expressive, it will be essential to examine whether the collected transcripts accurately represent these characteristics.

The dataset will be sourced from publicly available repositories containing real TikTok transcripts. These sources will be reviewed to ensure that the content is authentic and reflects the variety of expressions typically found on the platform. The structure of the transcripts will be examined to determine how users naturally present opinions, unsupported claims, or references to credible sources. This qualitative exploration will help determine whether the dataset contains the linguistic variation required for the project's credibility labels.

A quantitative assessment will also be conducted. This will include profiling the dataset to identify missing values, duplicates, inconsistent formatting, and any noise that may need to be addressed in later phases. Class distributions will be examined to observe whether certain credibility categories appear more frequently than others. Identifying these natural imbalances early will help shape decisions about preparation, balancing, and modelling.

Diversity within the data will also be assessed. Because TikTok covers a wide range of topics, the dataset will be reviewed to ensure that it includes content from multiple domains such as lifestyle, commentary, advice, and general discussions. A diverse dataset will support better model generalization.

By the end of this phase, a clear understanding of the data's suitability, limitations, and coverage will be established. These insights will guide the cleaning, labelling, and transformation steps planned for the next phase.

3. Phase 3: Data Preparation

The third phase will focus on preparing the raw transcripts so that they become clean, structured, and ready for modelling. Because TikTok text often includes emojis, informal expressions, inconsistent punctuation, and platform-specific elements, careful preparation will be required to ensure that the model learns from meaningful patterns.

The preparation process will begin by consolidating all transcript sources into a unified dataset. Once combined, structural issues such as missing values, duplicated entries, and formatting inconsistencies will be addressed. This step will ensure that the dataset is stable and free from errors that could influence the model.

After cleaning, the text undergoes standard preprocessing steps used in Natural Language Processing. These include:

1. Converting all text to lowercase
2. Removing punctuation
3. Tokenizing the text into meaningful units
4. Filtering out stop words that do not contribute to credibility distinctions.
5. Normalization techniques such as lemmatization may also be applied to ensure that different forms of the same word are treated consistently.

These transformations will ensure that the dataset is uniform and ready for feature extraction.

A key component of this phase will involve applying the credibility labels that the model will later learn to predict. Three predefined categories will be assigned to the transcripts: opinion, claim, and supported claim. These labels will be based on linguistic cues identified earlier in the study. Opinions will reflect subjective viewpoints, claims will include assertive statements without supporting evidence, and supported claims will reference credible sources.

Completing this phase will result in a clean, consistently formatted, and accurately labelled dataset. This prepared dataset will provide a strong foundation for the modelling experiments planned in the next phase.

4. Phase 4: Modelling

The modelling phase will explore a range of Natural Language Processing and Machine Learning techniques to determine the most suitable approach for credibility classification. Since the best method cannot be assumed beforehand, this phase will involve systematic experimentation.

The first step will involve converting the preprocessed transcripts into numerical representations. Multiple word-representation methods will be tested, such as frequency-based approaches and embedding techniques designed to capture semantic relationships. These different representations will help reveal which methods best capture the linguistic patterns needed for credibility assessment.

Several machine learning models will then be trained using these representations. Classical algorithms and more advanced methods will be explored to understand how each performs when dealing with short, dynamic social media text. Each model will be evaluated using consistent procedures to ensure meaningful comparison.

Hyperparameter tuning may also be carried out to refine model performance and identify promising configurations. The emphasis will be on understanding how each modelling technique behaves rather than selecting a final model prematurely.

By the end of this phase, the results of the experiments will guide the selection of the model that is most appropriate for the credibility classification task. The chosen model, along with its justification and performance results, will be presented later in the Fact-Finding and Results section.

5. Phase 5: Evaluation

The evaluation phase will focus on testing and validating the model that will be selected at the end of the modelling stage. Once the most suitable approach is identified, it will undergo a detailed assessment to ensure that its performance is reliable and consistent across all credibility categories. This testing process will help confirm whether the model is ready to be deployed as part of the final system.

The selected model will be evaluated using standard classification metrics such as accuracy, precision, recall, and F1-score. These metrics will provide insight into how well the model distinguishes between opinions, claims, and supported claims. Confusion matrices will also be examined to highlight areas where the model performs strongly and areas where misclassifications may occur.

In addition to numerical evaluation, misclassified transcripts will be reviewed to understand why certain predictions fail and what linguistic patterns may challenge the model. This qualitative analysis will help ensure that the model not only performs well statistically but also aligns with the nature of real TikTok communication.

By the end of this phase, the evaluation results will confirm whether the selected model meets the performance standards required for deployment. Any necessary refinements will be identified, ensuring that the model is stable, reliable, and ready to be integrated into the web-based credibility assessment platform.

6. Phase 6: Deployment

The final phase will focus on preparing the complete credibility classification pipeline for deployment within the web-based platform. Since the system is intended to process real TikTok transcripts, the pipeline will need to handle the entire flow of operations, starting from raw text input and ending with the final credibility prediction.

During this phase, the pipeline will be structured to include all essential stages: text cleaning, preprocessing, feature transformation, and classification by the selected model. Each component will be configured so that incoming transcripts can be processed in the same way they were prepared during the modelling phase. Ensuring that the live pipeline mirrors the training pipeline will be essential for maintaining consistent and reliable performance.

The selected model will then be exported in a deployment-ready format suitable for integration with the backend application. Compatibility checks will be carried out to confirm that the pipeline functions smoothly when embedded into the system and that it can respond correctly to real-time inputs. This will include verifying that the text cleaning steps operate as expected, that numerical features are generated correctly, and that the model produces outputs in a format that the platform can interpret.

Initial integration testing will be performed to ensure that the pipeline communicates properly with the web interface. Any adjustments required to preprocessing logic, pipeline flow, or output formatting will be addressed during this stage. The goal will be to ensure that the system can deliver fast, lightweight, and accurate credibility predictions once deployed.

By the end of this phase, the entire pipeline from raw text processing to final classification will be fully prepared for deployment. This marks the final step of the

CRISP-DM methodology and will enable the system to provide users with seamless, real-time credibility insights for TikTok transcripts.

1.6 Project Limitations

Although this project aims to establish a practical and reliable foundation for credibility classification on TikTok transcripts, several limitations are expected due to the scope, available resources, and time constraints:

- The system is limited to text-based analysis and does not account for visual, auditory, or contextual cues present in TikTok videos. Important elements such as tone, gestures, music, and visual framing cannot be interpreted through transcripts alone.

- The dataset is sourced from publicly available TikTok transcripts, which may not fully represent the diversity of global TikTok content. Variations in language style, cultural expression, and topic categories may therefore be underrepresented.

- The credibility labels used in the project rely on linguistic cues, which can sometimes appear ambiguous in short-form content. Statements may overlap between categories, making classification challenging even for human annotators.

- The model will be built using lightweight NLP and machine learning techniques rather than large-scale deep learning architectures. While this supports efficiency and deployment readiness, it may limit the system's ability to capture complex linguistic nuances such as sarcasm, implicit meaning, or subtle persuasive language.

- The system does not perform full fact-checking or external verification. It identifies whether a transcript contains subjective language, unsupported claims, or references to credible sources, but it does not confirm the accuracy of those references or validate the factual correctness of the claim.

- These limitations reflect the intended scope of the project and provide a foundation for potential enhancements in future work.

1.7 Target Audience

The system is designed for three primary groups who interact with TikTok content in different ways and whose decisions are influenced by the credibility of the information they encounter or endorse.

1. The first target audience includes everyday TikTok users who frequently rely on short-form videos as sources of entertainment, advice, and information. Many users encounter content that appears authoritative but is presented without evidence or context, making it difficult to distinguish credible information.


2. The second audience includes experts, professionals, and individuals with recognized subject-matter knowledge who may join the platform as verified contributors. These individuals can review and contextualize content, provide corrections, and help clarify whether certain claims align with trusted sources. Verification strengthens their credibility, making it easier for users to trust their insights. The system therefore facilitates an environment where knowledgeable contributors can counter misinformation more effectively, offer clear guidance, and help debunk misleading trends before they spread widely.


3. The third key audience includes brands, companies, and marketing teams that collaborate with TikTok influencers for promotional or partnership activities.

Together, these audiences benefit from a system that promotes transparency, improves trust, and encourages more informed decision making across the TikTok ecosystem.

1.8 Summary

This chapter outlined the motivation for developing a credibility classification system for TikTok transcripts and highlighted the challenges caused by the platform's rapid spread of mixed information. The project's objectives and research questions were defined to guide the development of a structured, explainable model capable of distinguishing opinions, unsupported claims, and supported claims. The scope and limitations of the project were also presented, clarifying the focus on transcript-based analysis and the use of linguistic cues to support transparency and trust.

The chapter identified the main audiences who will benefit from the system, including TikTok users, verified experts, and brands that evaluate influencer credibility. Together, these elements establish the foundation for the next chapter, which will review existing research on misinformation, NLP techniques, credibility assessment frameworks, and the classification of short-form social media content.

This prepares the groundwork for understanding the theoretical background and related studies that inform the design and methodology of the system.

Branding and System Naming

The system developed in this project will be referred to throughout the report under the name **Faktz**, a short and memorable term inspired by the word "facts." The name reflects the system's core purpose of helping users understand the credibility of fast-moving social media content without disrupting the natural flow of online expression. The stylized ending with the letter "z" aligns with the communication style and identity of Gen Z users, who represent a significant portion of TikTok's audience. The branding incorporates a clean, modern visual identity that symbolizes verification, awareness, and video-centered content. This includes minimalistic design elements and a color palette based on a distinctive blend of purple and teal, chosen to convey a balance between technological precision and user-friendly accessibility. Introducing the name at this stage ensures consistency across the report and strengthens the system's identity both as a technical solution and as an emerging digital tool.

Figure 4: Faktz Logo

Chapter 2: Literature Review

2.1 Overview

This chapter highlights the academic work that frames the study and establishes the theoretical foundations underpinning the credibility assessment of TikTok content. As short-video platforms continue to reshape how information is communicated and consumed, the literature highlights a growing concern regarding the accuracy and reliability of content shared online. Several studies emphasize that TikTok, in particular, has become a major source of information for younger audiences while simultaneously demonstrating a high vulnerability to misinformation. The aim of this literature review is to examine existing research on misinformation detection, NLP-based classification, credibility cues, and influencer–brand dynamics, and to identify the limitations and gaps that this project seeks to address. The chapter positions the research within this broader context and justifies the need for a system capable of distinguishing opinions, unsupported claims, and supported claims in TikTok video transcripts.

2.2 Area of the Study

2.2.1 Background

TikTok's rapid rise has transformed it into a global hub of information sharing, where short-form videos influence decisions on health, politics, lifestyle, and self-improvement. Studies show that young users increasingly rely on the platform for advice and knowledge, often accepting content as credible due to the informal and relatable nature of creators' delivery. Kirkpatrick and Lawrie (2024) [1] found that TikTok has become a primary source of health information for young women in the United States, yet a substantial portion of this content was rated as low quality or misleading. The short, fast-paced nature of TikTok videos encourages simplified explanations that frequently omit context or scientific grounding. This environment reinforces the importance of developing mechanisms that can help users evaluate the reliability of what they consume.

2.2.2 Factors

- TikTok algorithm prioritizes engagement, meaning sensational or emotionally charged content is amplified even if it lacks credibility.
- Creators often mix personal experiences with factual-sounding statements, which makes audiences interpret subjective views as objective information.
- The short-video format encourages oversimplification, reducing the depth and accuracy of explanations.
- Many videos appear authoritative despite lacking context, evidence, or verifiable references.

Together, these conditions produce an information ecosystem where distinguishing between accurate information and persuasive personal interpretation becomes challenging.

2.2.3 Previous Related Studies

The following papers highlight key research on misinformation detection, content credibility, and influencer–brand dynamics, outlining existing approaches, their limitations, and how they inform this project's aim.

1. **Pereira, B.B. and Ha, S. (2024). Environmental Issues on TikTok: Topics and Claims of Misleading**

   Pereira and Ha explored how environmental misinformation circulates on TikTok, focusing specifically on the way creators present their messages within short-form videos. Their thematic analysis showed that many videos framed misleading statements as though they were factual, even when they lacked proper evidence or credible sources. The study illustrated how persuasive TikTok content can be when it blends confident delivery with visually engaging formats [5].

   Although the findings were insightful, the approach relied entirely on manual labelling, which limits scalability. The study did not apply any automated methods such as machine learning or NLP, and it treated all misleading videos as a single category rather than separating opinions, unsupported claims, and supported claims. This creates a clear opportunity for more systematic, AI-driven approaches that address finer distinctions in credibility.

2. **Cools et al. (2024) – Modeling Offensive Content Detection for TikTok**

   Cools and colleagues examined TikTok through the lens of offensive and harmful comment detection. Using NLP and deep learning techniques, they demonstrated that text classification models can perform strongly even within TikTok's informal, slang-heavy environment. Their findings offered valuable evidence that AI models can adapt to the platform's linguistic style, which is often noisy and unstructured [6].

   However, the study focused only on comments rather than full video transcripts, and it did not address credibility or misinformation. The narrow scope limits its direct relevance to credibility assessment, yet the work remains important for showing that language-based classification is technically feasible on TikTok.

3. **MultiTec – Shang et al. (2025) – A Data-Driven Multimodal Short Video Detection Framework for Healthcare Misinformation on TikTok**

   Shang, Zhang, Deng, and Wang proposed a multimodal system that analyzed text, audio, and video features simultaneously to detect healthcare misinformation. By combining multiple modalities, the model achieved strong performance, particularly on vaccine-related content. The work demonstrated how misinformation can be identified more accurately when multiple signals are considered [7].

   Despite its strengths, the system required heavy computational resources and was limited to the healthcare domain. These constraints make real-time or large-scale deployment challenging. The study highlights the potential of multimodal AI but also the need for lightweight, scalable approaches suitable for general TikTok content.

4. **Sokolova & Kefi (2020) – Influencer Credibility and Purchase Intentions**

   This study examined how influencer credibility and authenticity shape user behavior and purchase decisions. The authors found that audiences place significant weight on perceived credibility, which directly affects engagement and trust. The findings underline how influential creators can be, especially when audiences rely on them for guidance or information [8].

   However, the study was survey-based and did not provide technical methods for evaluating credibility. It reinforces the importance of credibility in online environments but also points to the need for computational tools that measure it more objectively.

5. **Capitol Technology University (2022) – TikTok and War Misinformation**

   This article outlines how TikTok's rapid growth has made it a major source of misinformation, especially because short videos and charismatic creators often make unverified claims appear credible. It highlights that around one-fifth of videos contain some form of misinformation, including false health advice, conspiracy theories, and misleading political content. The platform's recommendation algorithm intensifies the issue by creating echo chambers where users repeatedly encounter similar misleading videos. The article also notes TikTok's efforts to address the problem through fact-checking partnerships, content moderation, and educational campaigns, while recognizing that misinformation cannot be eliminated entirely [9].

   This contextual perspective is relevant to the research because it illustrates the scale of the misinformation problem on TikTok and reinforces the need for automated credibility-assessment systems.

6. **Belanche et al. (2021) – *Understanding Influencer Marketing: The Role of Congruence Between Influencers, Products, and Consumers.***

   Belanche and colleagues examined how the alignment between influencers, products, and audiences affects marketing outcomes. They found that when an influencer's message does not match their typical identity or audience expectations, credibility decreases [10].

   This relates to credibility detection because it highlights how inconsistencies in messaging can signal unreliability, insight that can inform AI systems that assess the trustworthiness of TikTok content.

Table 2: Proposed Papers

Author (Year)	Title	Purposed Study	Outcome	Remark (Limitation / Future Work Suggestion)
Pereira & Ha (2024)	Environmental Issues on TikTok: Topics and Claims of Misleading Information	To analyze misleading environmental claims in TikTok videos through thematic classification.	Found that many creators deliver unsupported claims confidently, making them appear credible.	Entirely manual; no NLP or automation; does not separate opinions, unsupported claims, or supported claims. Suggests need for automated credibility cl
Cools et al. (2024)	Modeling Offensive Content Detection for TikTok	To detect offensive/harmful TikTok comments using NLP and deep learning.	Demonstrated that AI models perform well on TikTok's informal language.	Focused only on comments, not video transcripts; does not address misinformation. Shows technical feasibility for text classification on TikTok.
Shang et al. (2025)	MultiTec: A Multimodal Short Video Detection Framework for Healthcare Misinformation on TikTok	To detect healthcare misinformation using combined text, audio, and visual features.	Achieved high accuracy, especially for complex medical misinformation.	Computationally heavy; domain-specific; not scalable for general TikTok content. Suggests need for lightweight, generalisable models.

Sokolova & Kefi (2020)	Influencer Credibility and Purchase Intentions	To examine how influencer credibility shapes user trust and behaviour.	Found that credibility strongly influences engagement and purchase intention.	Survey-based; no computational method; highlights importance of credibility but lacks technical assessment approaches.
Belanche et al. (2021)	Understanding Influencer Marketing: Congruence Between Influencers, Products, and Consumers	To study how alignment between influencers and their content affects credibility and engagement.	Found that misaligned messages reduce user trust and credibility.	Behavioural focus; no computational system. Suggests content–source consistency as a useful credibility signal.
Capitol Tech University (2022)	TikTok and War Misinformation	To explain how misinformation spreads rapidly on TikTok due to platform design and algorithmic amplification.	Identified widespread health, political, and conspiracy misinformation; highlighted echo-chamber effects.	Not empirical research; lacks technical modelling. Reinforces need for automated credibility assessment tools.

2.3 Related Method

To support the selection of an appropriate methodological approach for this study, Table 3 summarizes the key strengths, drawbacks, and limitations of the main methods identified in the reviewed literature. This comparison highlights how each method contributes to misinformation research and clarifies why an NLP-based text classification approach is the most suitable for the proposed credibility assessment system.

Table 3: Related Methods

Criteria	**Method1:** Manual Thematic Analysis	**Method 2** NLP-Based Text Classification	**Method 3** Multimodal Deep Learning
Strength(s)	Provides rich qualitative insight; captures nuanced meaning in videos.	Scalable, efficient, and adaptable to large datasets; effective on TikTok text.	High accuracy by combining text, audio, and visuals; captures complex misinformation cues.
Drawback(s)	Extremely time-consuming; subjective; not scalable; inconsistent across coders.	Depends on transcript quality; may miss visual or audio cues beyond text.	Heavy computational cost; requires large datasets; domain-specific.
Limitation(s)	cannot support real-time detection or large-scale analysis.	Limited to text-only features; may not detect multimodal misinformation.	Not generalizable across all TikTok topics; impractical for lightweight deployment.

Method Finding:

By reviewing the strengths and limitations of the existing methods, this project was able to identify what works well in current approaches and what gaps still remain.

Manual thematic analysis showed the value of understanding how creators frame misleading content, which helped emphasize the importance of separating opinions, unsupported claims, and supported claims. At the same time, multimodal studies demonstrated how rich TikTok videos can be, but also revealed that analyzing audio, visuals, and text together can be too demanding for broad, real-world use. Learning from both ends of the spectrum, this project focuses on transcript-based NLP because it offers a practical middle ground. It is scalable, efficient, and capable of capturing meaningful linguistic signals without the heavy computational cost of full multimodal systems. This makes it a more adaptable and realistic approach for building a credibility-assessment model that can be deployed on a scale.

2.4 Existing System / Application

When reviewing the systems currently used to manage misinformation on TikTok, it becomes clear that most of the platform's safeguards still depend heavily on human judgement rather than automated credibility assessment. TikTok combines automated moderation with a broad network of independent fact-checking organizations, who collectively support the review of questionable content across more than 60 markets. These partners include the Australian Associated Press (AAP), Agence France-Presse (AFP), Animal Político, Code for Africa (etc.) Their role is to verify claims and help TikTok decide whether certain content should be removed, labelled, or restricted within the For You Feed. Although this network is extensive, the process only begins once a video has been flagged or reported, meaning misleading content can circulate widely before it is reviewed [11].

TikTok's own misinformation policy reinforces this limitation. The platform notes that it does not allow misinformation that could cause harm and may add warning labels or remove videos entirely if fact-checkers determine that a claim is false. However, all of these actions occur after the content has already spread. There is no automatic evaluation of transcript credibility or claim strength while the user is watching the video. Users are therefore left without real-time support in judging whether the information they are encountering is reliable [12].

Initiatives outside of TikTok also contribute to combatting misinformation but face similar constraints. One example is MediaWise, run by the Poynter Institute, which collaborates with TikTok to debunk viral falsehoods and educate audiences about how to recognize misleading content. Although the educational value of such work is clear, it still relies entirely on human fact-checkers. This approach restricts its ability to keep pace with the enormous volume of TikTok uploads, and it does not offer users an automated assessment of individual claims.

TikTok additionally maintains a public Fact Check Center, which provides general explanations about common misinformation narratives. While this resource helps raise awareness, it does not evaluate specific videos or provide personalized credibility guidance. Viewers still must interpret claims on their own, even when videos appear highly convincing due to strong visuals, confident delivery, or emotional storytelling.

In academic research, systems like MultiTech illustrate what is technologically possible. MultiTech shows that multimodal AI can detect certain types of misinformation by analyzing text, audio, and visuals together. These systems perform well in specialized domains such as healthcare, but they are computationally heavy and not suitable for everyday use. They also require significant training data and do not generalize easily to the wide range of topics found on TikTok.

Outside the fact-checking and AI space, Reddit offers a different perspective that is relevant to this work. Reddit is built around open discussion, where users share experiences, opinions and stories, and credibility often emerges through community questioning, debate, and the natural back-and-forth of comments. This model of transparent discussion helped shape the idea behind Faktz, especially the focus on surfacing how different kinds of claims are interpreted by others. However, Reddit still lacks automated credibility detection and remains dependent on subjective community judgement.

Overall, existing systems either work reactively, require substantial human effort, or focus only on narrow domains. None provide a lightweight, transcript-based tool that automatically distinguishes between opinions, unsupported claims and supported claims in real time. This gap emphasizes the need for an application like Faktz, which aims to give users immediate, AI-generated clarity about the credibility of the TikTok content they encounter.

Table 4: Existing Systems and Proposed System

Features / Requirements	TikTok Fact-Checking	MediaWise	MultiTec	Faktz
Automated transcript analysis	X	X	✓ (partially, only within multimodal pipeline)	✓
Real-time credibility classification	X	X	X	✓
Opinion / Unsupported Claim / Supported Claim categories	X	X	X	✓
Accessible to general users	X (only internal moderation)	✓	X (research model only)	✓
Lightweight and scalable	✓ (moderation scale only)	X	X	✓
Domain-independent	X	✓	X (health-specific)	✓
Multilingual capability	✓ (depending on region)	X	X	X

2.5 Gap in Literature

Although existing studies provide valuable insight into misinformation, credibility, and influencer dynamics on TikTok, several important gaps remain unaddressed:

1. Current research tends to focus on either specific misinformation domains or isolated aspects of content analysis, leaving the broader challenge of general credibility assessment largely unexplored. For example, Pereira and Ha (2024) relied entirely on manual labelling and treated all misleading videos as a single category, without distinguishing between opinions, unsupported claims, and evidence-based statements. Basch et al. (2021) showed that health misinformation is widespread on TikTok, yet the study did not propose computational approaches for detecting false or misleading claims. Meanwhile, Cools et al. (2024) demonstrated that TikTok text can be processed effectively using NLP but did not extend this capability to credibility or misinformation contexts.

2. Multimodal research such as MultiTec (Shang et al., 2025) has achieved strong results in healthcare misinformation detection, but its domain-specific nature and heavy resource requirements limit its scalability and general use across TikTok content.

3. Studies on influencer credibility (Sokolova & Kefi, 2020; Belanche et al., 2021) highlight the social and behavioural importance of trust, yet they do not offer technical frameworks for evaluating credibility in real time or within algorithm-driven environments like TikTok.

4. Contextual analyses such as Capitol Technology University (2022) illustrate the platform-wide prevalence of misinformation but do not provide actionable computational methods for addressing it.

Collectively, these studies show that while misinformation on TikTok is well documented, there is no existing system that automatically classifies credibility categories such as opinion, unsupported claim, and supported claim using scalable NLP techniques. Current literature either depends on manual coding, examines only narrow aspects of language or behavior, or focuses on highly specialized domains. As a result, there remains a clear need for a lightweight, generalizable, and automated approach that can evaluate the credibility of TikTok content based on textual transcripts. Addressing this gap would contribute significantly to misinformation research and offer practical support for users, fact-checkers, and digital platforms.

2.6 Summary

This chapter brought together the main studies, methods and existing systems that shape our understanding of misinformation on TikTok. Across the literature, a consistent pattern appeared: misleading content spreads easily on the platform, especially on topics like environmental issues, health advice and politically sensitive events. Many studies showed how quickly unsupported claims can seem convincing when presented through short, engaging videos. However, most of this work relied on manual analysis or focused on narrow domains, which limits how far these findings can be applied in real-world settings. This highlighted the need for more scalable and automated approaches that can handle the fast pace of TikTok content.

The chapter also compared several methodological approaches, ranging from manual coding and text-based NLP techniques to more advanced multimodal systems. Each method offered something valuable, yet none provided a complete solution. Manual approaches helped reveal how creators frame misleading messages but cannot operate at scale. Multimodal systems demonstrated strong technical performance, but they are too resource-heavy for broad deployment. Through this comparison, it became clear that a transcript-based NLP approach offers a practical balance: it is efficient, adaptable and capable of analyzing linguistic signals without requiring the computational cost of full audio-visual processing.

When examining existing systems, the gap became even more apparent. TikTok's moderation tools and wide network of fact-checking partners play an important role, but they act mainly after misleading videos have already circulated. Educational efforts, such as those by MediaWise, help raise awareness but cannot keep up with the volume of new content. Even advanced academic systems like MultiTec remain limited by domain specificity and high resource requirements. Platforms like Reddit provided inspiration for open conversation, but they rely entirely on community judgement rather than automated credibility assessment.

Altogether, these findings point to a clear research need: there is currently no lightweight, accessible tool that can evaluate TikTok content in real time and distinguish between opinions, unsupported claims and supported claims. This gap strongly supports the motivation for developing Faktz, a system designed to offer users immediate, AI-driven insights into the credibility of the videos they encounter. The next chapter builds on this foundation by outlining the system's design and the methodology used to develop it.

Chapter 3: Research Methodology

3.1 Overview

This chapter explains the approach used to develop Faktz and describes how the system gradually evolved from a research idea into a working platform. The purpose of Faktz is to assess the credibility of TikTok content using Artificial Intelligence. The work presented here builds on the findings from Project Documentation Part I and focuses on how the system was organized, designed, and implemented.

Faktz relies on Natural Language Processing and Machine Learning to study the transcript taken from TikTok videos. After processing the transcript, the system places the content into one of three groups: Opinion, Claim, or Supported Claim. When a user submits a TikTok link, the platform extracts the text, analyzes it, and returns the assigned category along with a confidence value that reflects how certain the model is about its decision. For content identified as a Supported Claim, the system also presents the source mentioned in the transcript, allowing users to understand where the information may have come from.

The platform serves different audiences. Regular viewers can use it to check how credible a video might be. Professionals can contribute accurate information through their participation on the platform. Brands can use the credibility results to review influencers before considering them for partnerships.

To ensure that the system responds to real needs, this chapter also describes the fact-finding activities that guided the design, the requirements formed from user feedback and data analysis, and the structure that connects the ML model to the web interface. The work follows the CRISP DM methodology, which organizes the process into clear stages ranging from understanding the problem to preparing data, training models, evaluating results, and deploying the final system.

The chapter also includes several design visuals, such as the rich picture, use case, activity, class, and sequence diagrams, along with the early user interface designs. Together, these elements show how the system functions and how its different parts work together to deliver clear and accessible credibility assessments for TikTok content.

3.2 Fact-finding

3.2.1 Technique 1: Observation

Observation was used as the first fact-finding technique to gain direct insight into how TikTok content is created, presented, and consumed. This technique focused on analyzing real interactions on the platform, including the nature of videos that present advice, claims, or factual statements, and the behavioral patterns of users who view such content. The observations included examining the delivery style of creators, the persuasive cues used in videos, and the engagement dynamics reflected through comments, shares, and likes.

This method allowed a natural view of how misinformation or unsupported claims emerge and spread within everyday interactions. Observing user reactions in the comments section also provided valuable information about how audiences interpret content, whether they question its accuracy, and how easily misleading information can circulate.

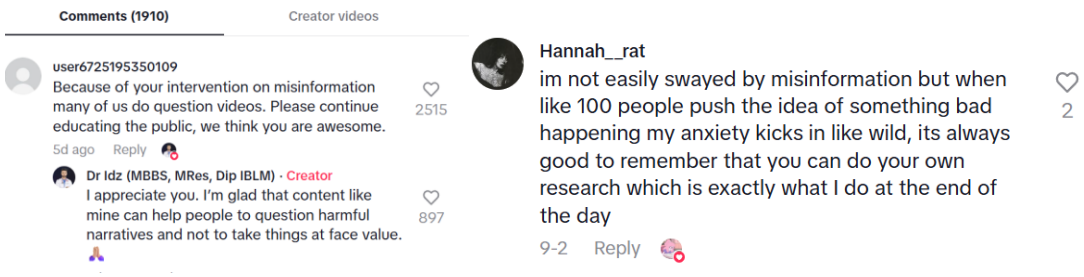


Figure 5: Comments from Tiktok showing users perception on misinformation

Findings:

This technique showed that many popular videos present advice, claims, or “facts” in a confident and engaging style, especially in areas such as health, fitness, lifestyle, and politics. This delivery often makes unsupported statements appear credible. The comments analyzed (Figure 5) demonstrated that users frequently respond with agreement or emotional reactions rather than verification, indicating that information is often accepted at face value.

The comment sections also revealed that some users are aware of misinformation risks yet still feel influenced when many viewers reinforce a particular claim. This behavior reflects how quickly misinformation can spread through engagement rather than accuracy. These observations highlight the need for a tool that provides immediate and accessible credibility indicators to support users in evaluating content more critically.

3.2.2 Technique 2: Questionnaire

A structured questionnaire was used to gather quantitative data on user behavior, exposure to misinformation, and expectations for credibility-checking tools. The questions explored how frequently respondents use TikTok, the types of content they engage with, how often they verify information, and their experiences with misleading videos. Additional sections examined reasons behind misinformation, attitudes toward influencer responsibility, and perceptions of AI-assisted analysis.

The questionnaire also assessed user's willingness to trust a platform like Faktz, including the usefulness of confidence scores and the value of expert involvement. This approach ensured coverage of both behavioral patterns and user needs.

The full list of questionnaire items, including all sections and response options, is provided in Appendix A to ensure clarity and transparency of the data collection process.

Findings & Analysis:

The questionnaire findings provided a detailed view of how respondents engage with TikTok and how this engagement shapes their exposure to potential misinformation. A substantial majority reported using the platform on a daily basis, which significantly increases the likelihood of encountering advisory or claim-driven content throughout the day. This is particularly notable given that TikTok's algorithm prioritizes rapid, high-engagement videos, meaning that users are continually exposed to material that appears informative yet may lack verifiable evidence. As shown in the figures attached below, the categories most frequently consumed (health, fitness, lifestyle, and politics) are precisely those where misinformation tends to spread quickly due to the blend of personal opinions, persuasive delivery, and trending narratives.

The responses revealed that users rarely engage in verification practices before accepting or sharing information, a finding that aligns with observed patterns across social media ecosystems. Most participants indicated that they "rarely" or "never" check the accuracy of claims presented in videos. This behavior contributes to an environment where misleading content can circulate widely with very minimal resistance. A notable proportion of respondents also admitted that they had previously believed a video that later turned out to be false, highlighting the platform's strong persuasive influence, particularly when content is delivered confidently or reinforced by large audiences.

When asked to reflect on why misinformation spreads so easily, participants commonly pointed to motivations such as the pursuit of higher engagement, sponsorship incentives, and a general lack of subject knowledge among creators. These perspectives demonstrate public awareness of the structural factors that shape online content production. Despite these concerns, respondents expressed willingness to rely on AI-assisted tools for credibility assessment, provided that the system offers clear explanations, transparent confidence scores, and accessible interpretation. The involvement of verified professionals was also seen as an important factor in enhancing user trust and promoting responsible dialogue.

Collectively, these findings confirm that users are not only exposed to large volumes of potentially misleading content but also lack reliable mechanisms for assessing credibility in real time. They also reveal strong support for tools that enhance clarity and promote critical engagement. All related figures that illustrate these findings are attached below for detailed reference and visual analysis.

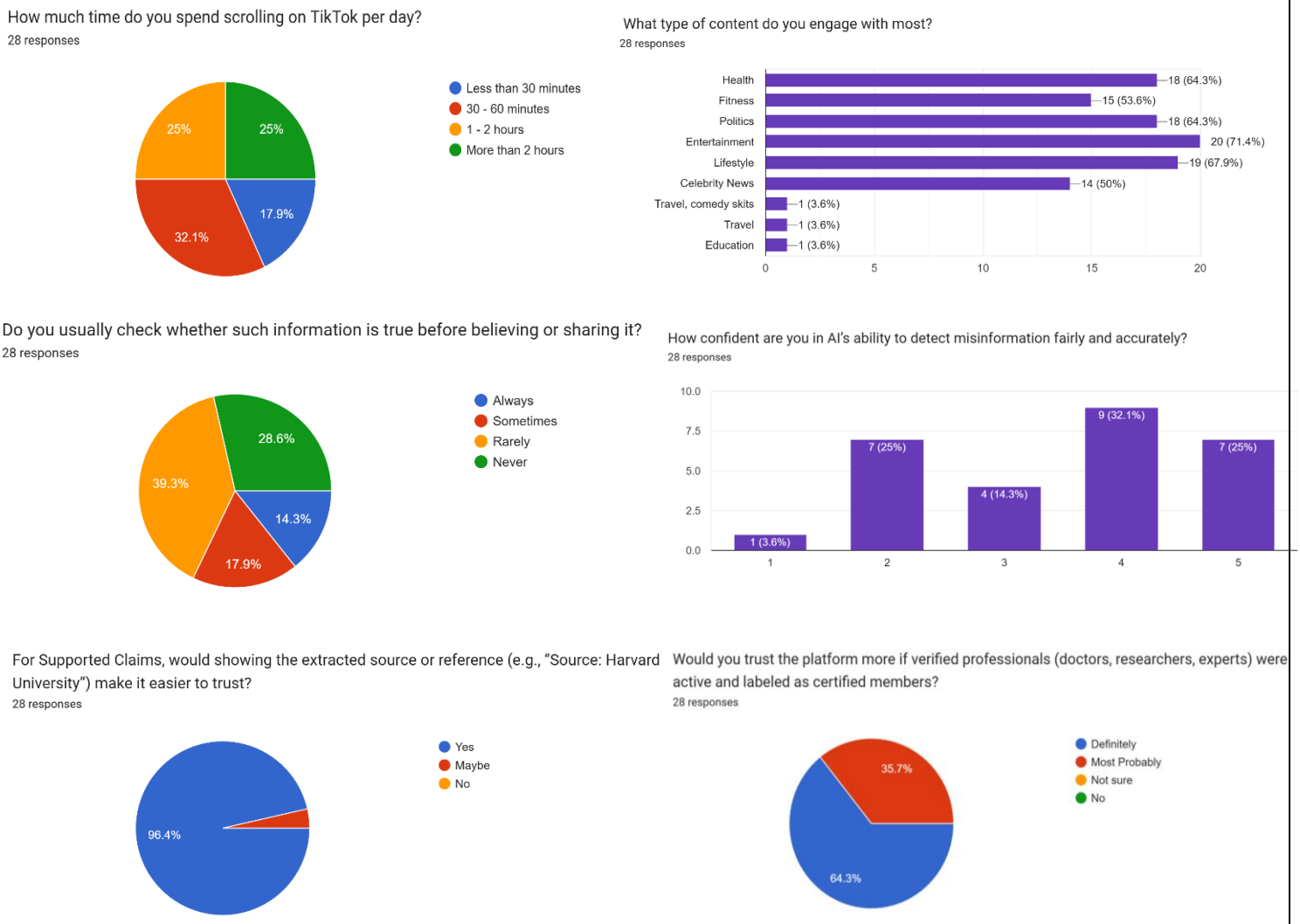


Figure 6: Questionnaire Results

3.2.3 Technique 3: Dataset Collection

To support the development of the Faktz AI credibility classification model, multiple datasets were collected, merged, and refined to ensure both linguistic diversity and representativeness of online misinformation patterns. The goal was to create a dataset that accurately reflects how people express opinions, make claims, and reference evidence on TikTok.

A. Data Sources

The final dataset was constructed using a combination of three main data sources:

1) **TikTok Transcript Dataset (Kaggle)**

This dataset contained TikTok video transcriptions and basic metadata such as likes, views, and captions. The text samples were short, informal, and often included statements of opinion or advice, making them suitable for real-world examples of social media discourse.

Initially, it only contained two categories , *Claim* and *Opinion*. Additional manual annotations were performed to introduce a third category, *Supported Claim*, by identifying statements that referenced credible institutions or sources (e.g., “According to WHO” or “as Harvard research shows”).

Data source: Ross, A. (2023). tiktok video transcipts. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/antoineross/tiktok-video-transcipts.

2) **Real and Fake News Dataset (Kaggle)**

To enhance the variety and depth of language patterns, a second dataset titled “Real and Fake News” by Razana Qvi was integrated. This dataset contained more than 20,000 labeled articles categorized as Real or Fake. Although its format and length differed from TikTok captions, its inclusion helped the model learn how factual and misleading information are linguistically structured. The combination of short-form social content and long-form articles provided a balanced dataset for training the classifier to recognize credibility features across contexts.

Data source: Raza, A. (2025). *Real & Fake News*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/razanaqvi14/real-and-fake-news.

3) **Synthetic Dataset (Custom Generated)**

A third synthetic dataset was generated to address class imbalance and introduce more topic diversity. Using text templates and topic prompts, additional examples were created for categories such as health, beauty, finance, and social issues. Each generated entry simulated the tone and structure of TikTok posts while preserving realistic grammar and language. This helped the model generalize better to unseen data and reduced bias caused by uneven class representation.

B. Data Merging and Organization

The collected datasets were merged into a single structured dataset with three planned categories: Opinion, Claim, and Supported Claim. Each entry contains a short text segment representing one of these classes. The merging process involved aligning column structures, removing duplicates, and ensuring consistency in class labeling across sources.

The combined dataset resulted in approximately 45,000 total samples, distributed across the three planned categories. As shown in Figure 5, Claim entries formed the largest portion (19,049 samples), followed by Supported Claim (15,121) and Opinion (9,892). This slight imbalance reflects real social-media trends, where users tend to make assertions more frequently than they share evidence-based or personal reflections.

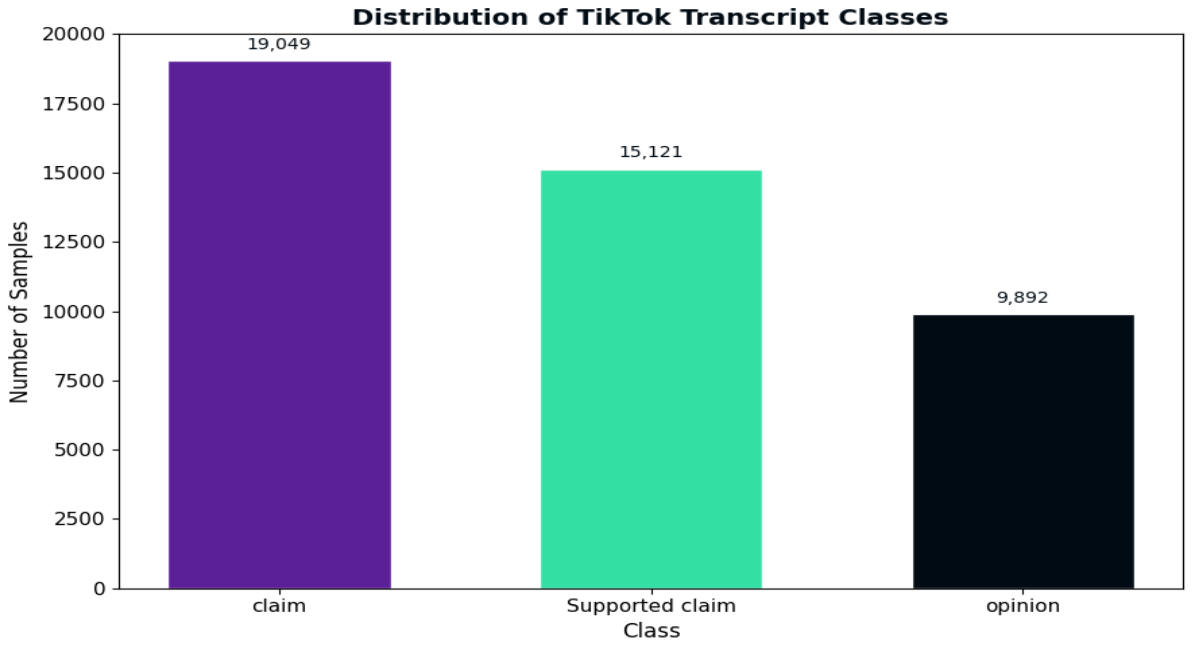


Figure 7: Distribution of TikTok Transcript Classes

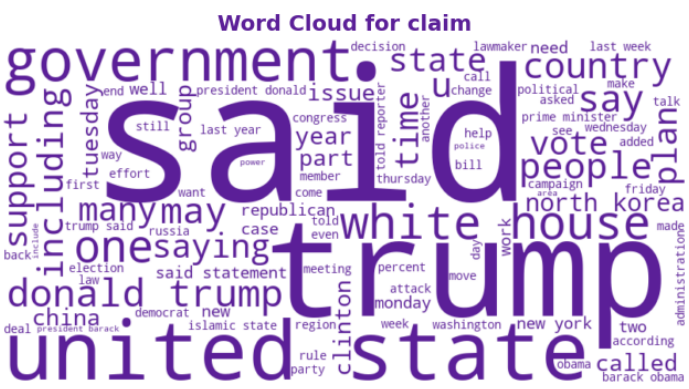
A sample of the merged dataset is provided in Appendix B for reference.

C. Preliminary Data Exploration

To better understand the collected text, word clouds were created for each class.

The figures above show clear differences in language style between categories.

- Claim entries commonly include assertive or news-related terms such as “said,” “government,” and “Trump.”
- Opinion entries emphasize subjective expressions like “willing,” “think,” and “view.”
- Supported Claim entries often mention organizations or leaders, indicating factual references (e.g., “United States,” “president,” “report”).



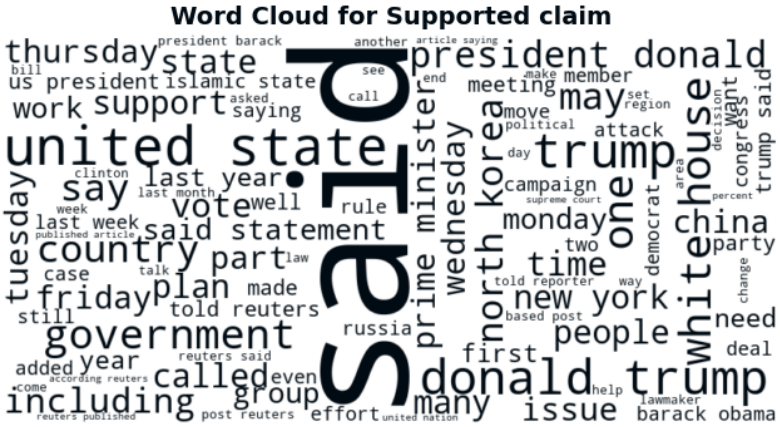


Figure 8: Word Cloud for data classes

3.3 System Requirement Analysis

The requirements for the Faktz system were shaped by the insights gathered during the fact-finding stage, which included both the user questionnaire and the analysis of the collected datasets. The questionnaire responses showed that many TikTok users regularly come across misleading content, often believe it, and rarely take time to verify its accuracy before sharing it. Participants also demonstrated a strong interest in AI tools that offer quick, clear, and reliable credibility indicators. These findings guided the development of the functional and non-functional requirements, ensuring that the system remains practical for users while also being technically achievable.

The dataset analysis provided additional direction for the design of the system. The text gathered from TikTok transcripts and online news sources displayed a wide range of writing styles, tones, and structures. Many samples included informal expressions, emojis, links, punctuation, and other forms of noise. Because of this, the model needs to perform several preprocessing tasks before classification can take place. These tasks include removing irrelevant elements, standardizing the text, and converting it into a numerical structure that can be processed by Artificial Intelligence techniques. These steps are essential for improving prediction quality and helping the model interpret real world social media text more accurately. The following subsections outline the functional and non-functional requirements derived from these findings.

3.3.1 User Requirements

Users require a platform that is fast, reliable, and easy to navigate, with credibility assessments that enhance rather than disrupt their normal social-media habits.

- Users need a quick and easy way to check the credibility of TikTok videos.
- Users need clear and understandable credibility labels without technical complexity.
- Users need confidence scores or brief explanations to help them interpret the results.
- Users need a simple interface that allows them to submit links and view results effortlessly.
- Users need visible verification for professionals so they can trust expert input.
- Users need accessible community features to discuss and respond to credibility assessments.
- Users need a safe environment where discussions remain respectful and reliable information is encouraged.

Stakeholders

1. **General Users:** Individuals who use the platform to check the credibility of TikTok videos and participate in discussions.
2. **Content Creators:** TikTok creators whose videos may be analysed on the platform and who may benefit from improved credibility and visibility.
3. **Verified Professionals:** Experts such as doctors, researchers, and specialists who contribute authoritative insights and enhance trust through verified profiles.

3.3.2 Functional Requirements

The functional requirements define the main operations that the Faktz system/model must perform to achieve it's intended purpose.

Table 5: Functional Requirements

#	Requirement	Description
1	Data Ingestion	The system should accept text input from various sources, such as TikTok video transcripts, captions, or written statements, for credibility analysis.
2	Data Cleaning and Preprocessing	The system should remove irrelevant elements (e.g., emojis, URLs, punctuation, stopwords) and standardize the text (lowercasing, lemmatization) before analysis.
3	Text Representation	The system should convert the cleaned text into a numerical format using text representation techniques (e.g., TF-IDF or vector embeddings).
4	Credibility Classification	The system should classify each text sample into one of three credibility categories: Opinion, Claim, or Supported Claim.
5	Confidence Scoring	The system should output a confidence score indicating the AI model's certainty level for each classification.
6	Source Extraction for Supported Claims	When a Supported Claim is identified, the system should extract and display the referenced source.
7	Result Presentation	The system should present results in a clear, user-friendly format displaying the label, confidence score, and extracted source (if applicable).
8	Integration Capability	The system should be designed with an API-ready structure to allow integration with the Faktz web platform.

Summary: These requirements ensure that Faktz can process text efficiently, classify it accurately, and communicate results in a transparent and user-centered manner. Together, they reflect both user expectations (clarity, trust, simplicity) and technical needs (clean data, explainable AI, scalability).

3.3.3 Non-Functional Requirements

The non-functional requirements define quality standards and performance

goals, and ethical considerations that the Faktz system must meet.

Table 6: Non-Functional Requirements

#	Requirement	Description
1	Accuracy and Consistency	The system should achieve reliable classification accuracy across all categories to ensure that results are consistent and trustworthy.
2	Performance and Speed	The system should process and return results efficiently, ideally within a few seconds per text input, reflecting the fast-paced nature of social-media use.
3	Transparency and Explainability	The model's decisions should be interpretable through the inclusion of confidence scores and visible references for supported claims. This directly addresses users' trust concerns identified in the questionnaire.
4	Usability and Simplicity	The user interface and outputs should be intuitive, concise, and visually clear so that users of different technical backgrounds can understand credibility outcomes.
5	Data Privacy and Ethics	The system should not store personally identifiable information. The model should promote awareness rather than censorship, maintaining ethical neutrality.
6	Scalability	The system should be scalable to support larger datasets, new content domains, and future extensions.
7	Maintainability	The system should be designed for easy maintenance, allowing updates to the dataset, retraining of the model, and replacement of components without disrupting performance.
8	Reliability and Error Handling	The system should detect and handle incomplete or invalid text input gracefully, preventing crashes and ensuring stable performance under different conditions.

3.4 System Design

3.4.1 Rich Picture Diagram

The rich picture diagram provides an overview of how misinformation spreads across TikTok and how the Faktz platform interacts with the different stakeholders involved. It shows that content creators may unintentionally share misleading information, which affects their credibility and influences users who often struggle to determine what is reliable. Faktz acts as an intervention point by analyzing TikTok videos, assigning credibility labels and confidence scores, and helping users make more informed judgments.

Verified experts contribute by reviewing content, offering corrections, and strengthening trust through professional badges. The diagram also illustrates the impact on brands, which rely on credible creators for safe collaborations. By helping brands identify trustworthy influencers, Faktz supports more responsible content partnerships. Overall, the diagram highlights the ecosystem of misinformation and positions Faktz as a tool that promotes clarity, trust, and informed engagement.

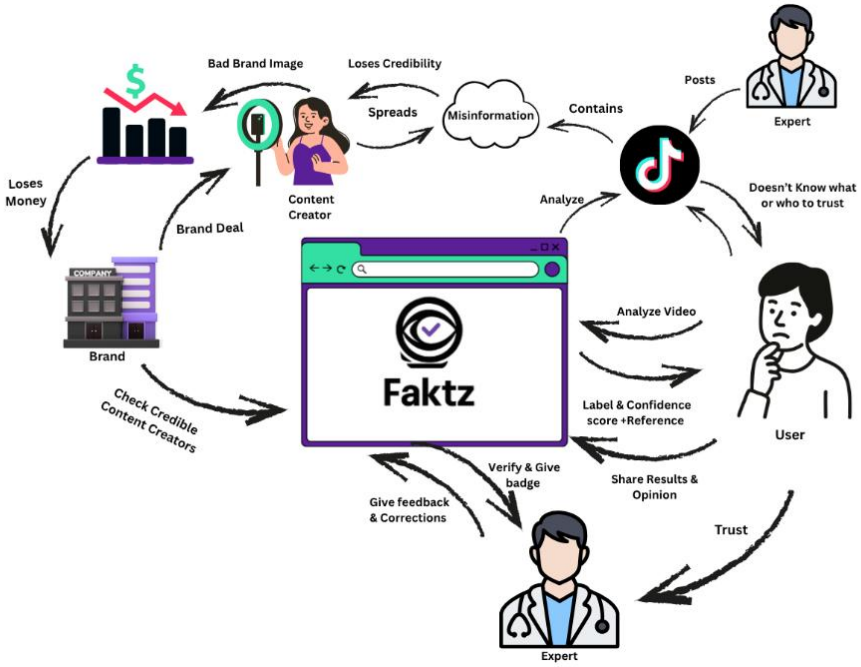


Figure 9: Rich Picture Diagram

3.4.2 Use case diagram

3.4.2.1 List of actors

Table 7: List of Actors

Actors	Relationship	Use Case	Use Case Description
Normal User, Expert User	–	**Login / Register**	Allows users to create an account or log in to access system features.
Normal User, Expert User	*Extends*	**Edit Profile**	Lets users update their personal information such as name, bio, or contact details.
Normal User, Expert User	*Extends*	**Delete Profile**	Enables users to permanently remove their account and related data.
Expert User	*Includes*	**Submit Profession Credentials**	Expert users upload documents to verify their professional background.
Expert User	*Includes*	**Request Expert Verification**	The system reviews submitted credentials and verifies the expert's profile.
Normal User, Expert User	*Includes*	**Submit TikTok Video Link**	Allows users to input TikTok video URLs for AI analysis.
System (Automatic)	*Includes*	**Classify Video**	The AI model analyzes the video transcript and classifies it.
System (Automatic)	*Includes*	**Display Results**	Shows the AI-generated credibility classification and explanation.
Normal User, Expert User	*Extends*	**Post Results**	Allows users to share the AI classification results publicly on the platform.
Normal User, Expert User	–	**Comment & Engage with Posts**	Enables users to comment, like, and participate in discussions related to posted results.

3.4.2.1 Use Case Description

The use cases listed below represent the features that fall within the scope of this project. Authentication functions and CSI-related system features are handled separately and are included in my colleague's report.

Use Case	Submit TikTok Video Link	
Precondition	The user is logged in and has access to the analysis page.	
Typical Course of events	**Actor Action**	**System Response**
	User enters a TikTok video link into the input field.	System validates the link format.
	User submits the link for analysis.	System sends the link to the API service and retrieves the transcript.
		System processes the transcript, classifies the content, and prepares the results for display.

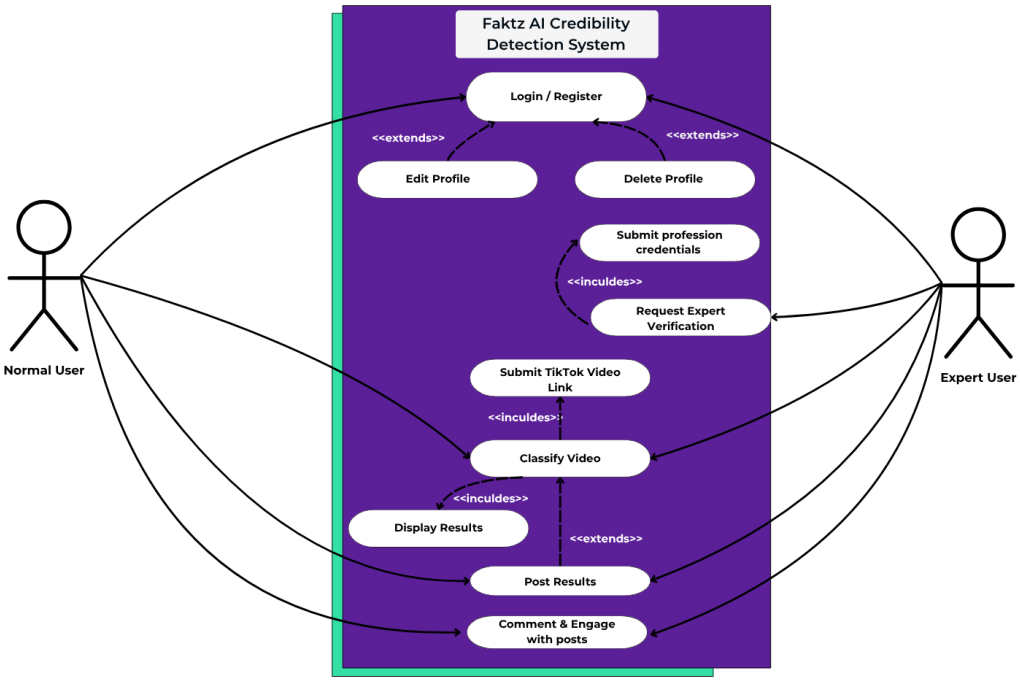


Figure 10: Use-Case Diagram

3.4.3 Activity Diagram

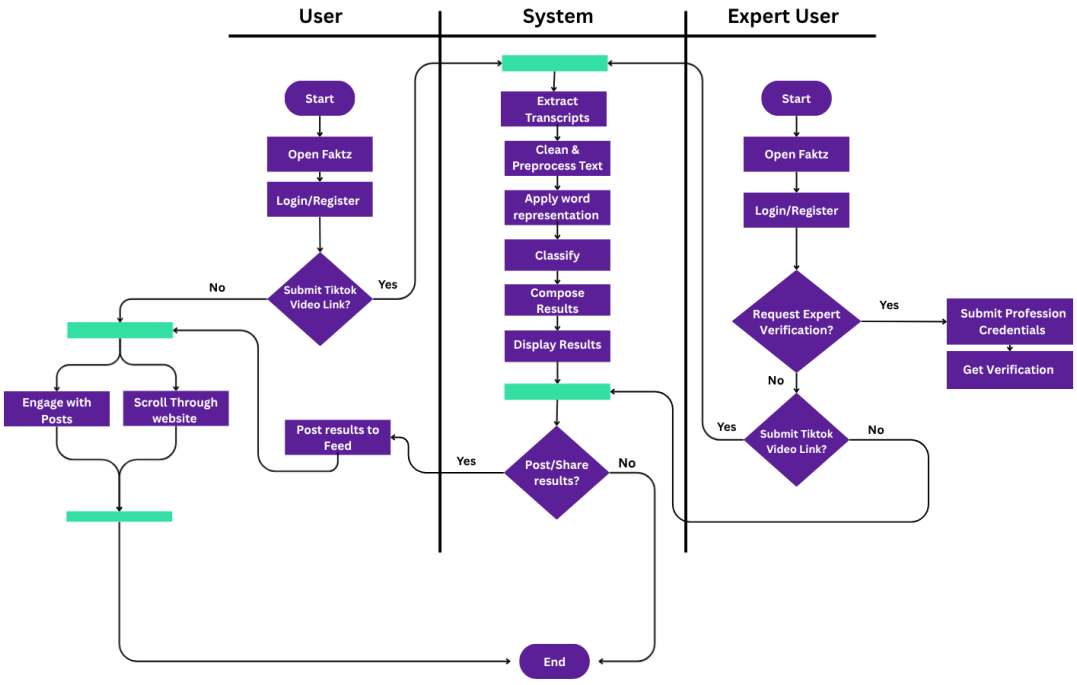


Figure 11: Activity Diagram

The activity diagram outlines the main workflow of the Faktz platform, focusing on the core functions covered in this project. Users begin by accessing the system and submitting a TikTok video link for analysis. The system then extracts the transcript, preprocesses the text, applies word representation, and classifies the content before displaying the credibility label and confidence score. After receiving the results, users can choose to post them to the community feed, where others can view and engage with the content.

For expert users, the diagram includes an additional path that allows them to submit professional credentials and request verification. Once verified, they can provide authoritative feedback on posts. Overall, the diagram captures how users interact with the platform from analysis to community participation.

3.4.4 Sequence Diagram

The sequence diagram demonstrates how the main components of the Faktz credibility-analysis system interact when a user submits a TikTok video link for classification. The process begins when the API receives the video link and forwards it to the Flask application. The Flask service then extracts and preprocesses the transcript, removing noise and standardising the text. Once cleaned, the text is passed to the vectorizer, which converts it into a numerical representation suitable for model processing.

The model then performs the credibility classification and returns a label along with a confidence score. If the output label is “Supported Claim,” the system triggers the SourceExtractor component to identify and extract any referenced sources. Otherwise, the source field remains empty. All outputs are then compiled and returned through the API to be displayed to the user.

Overall, the diagram outlines the end-to-end flow of data through the system, showing how each component contributes to transforming raw TikTok text into structured, interpretable credibility results.

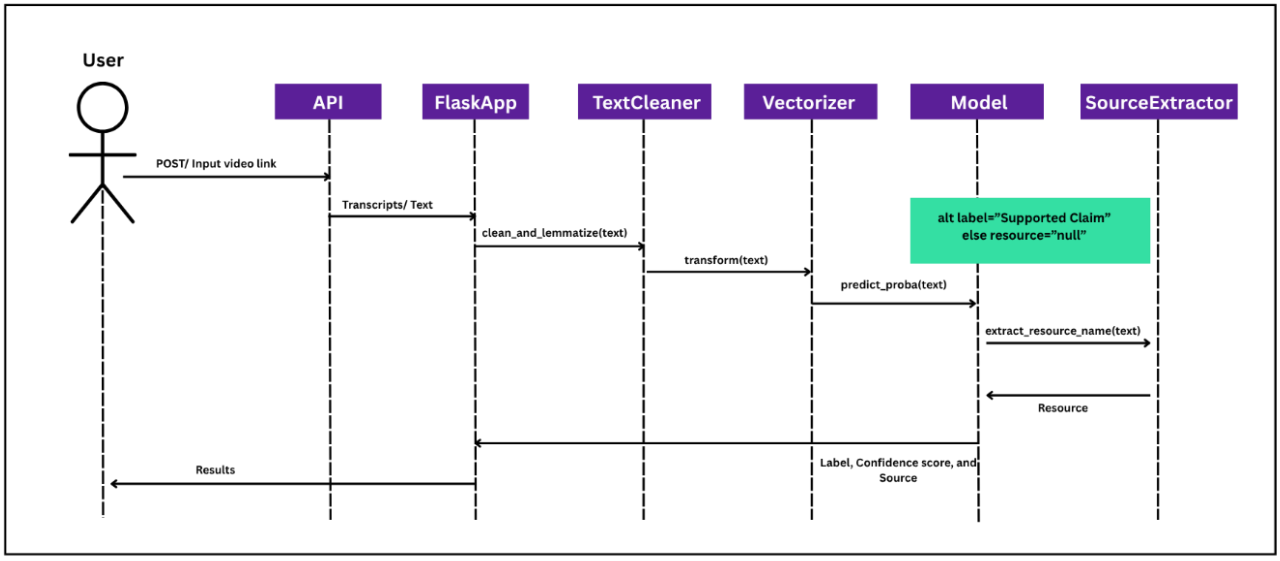


Figure 12: Séquence Diagram

3.4.5 User Interface Design

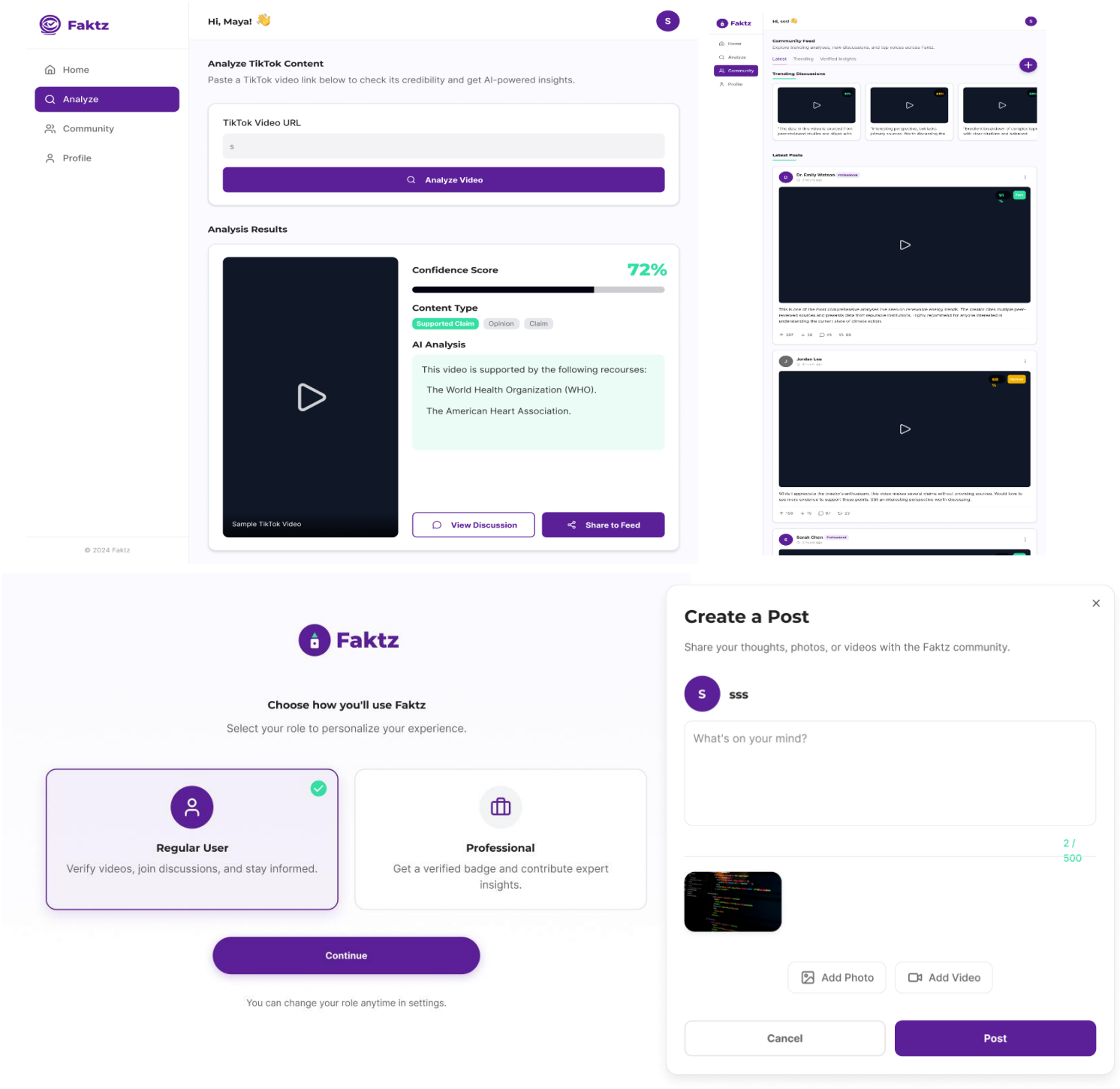


Figure 13: User Interface Design

3.4.6 System Flow Diagram

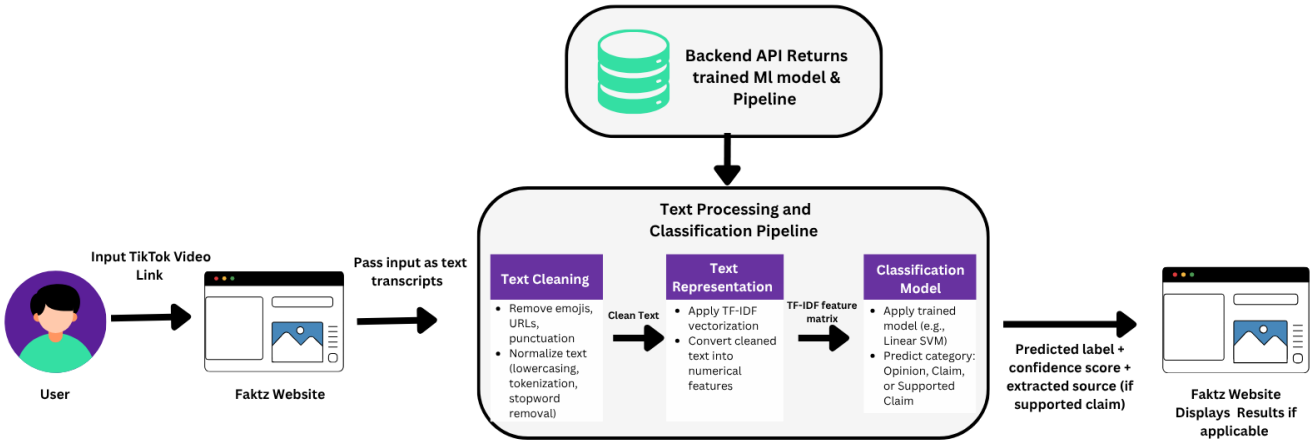


Figure 14: System Flow Diagram

3.5 Summary

Chapter 3 brought together all the methods and design decisions that shaped the development of the Faktz system. The fact-finding stage played an essential role in grounding the project in real user behavior. By observing TikTok content, gathering feedback through the questionnaire, and analyzing datasets from Kaggle, it became clear how easily misinformation spreads and how much users rely on quick, understandable tools to help them judge credibility. These insights guided the direction of the system and helped define realistic user, functional, and non-functional requirements.

The second part of the chapter focused on turning these requirements into a clear design. The rich picture diagram captured the wider environment in which Faktz operates and showed how different groups, from everyday users to experts and brands, interact with misinformation online. The use case diagram and descriptions helped clarify what each type of user can do on the platform, while the activity and sequence diagrams mapped out the internal flow of actions and processes, from submitting a TikTok link to receiving a credibility result. The system flow diagram added a high-level view of how data moves through the platform, and the user interface designs showed how these ideas translate into a functional and accessible experience.

Overall, this chapter established the structure, logic, and purpose of the system, providing a clear foundation for the implementation that follows.

Chapter 4: System Design

4.1 Model Information

The Faktz system integrates a machine learning model designed to classify TikTok video transcripts into three categories: Claim, Supported Claim, and Opinion. The model was developed after an extensive evaluation of multiple algorithms such as Logistic Regression, Random Forest, and Support Vector Machines. Through comparative testing, the TF-IDF vectorization technique combined with a Linear Support Vector Machine demonstrated the highest performance in terms of accuracy, generalization ability, and execution speed. To ensure reliable confidence scoring, the classifier was wrapped with a calibration layer, enabling the system to produce probability-based outputs that support later credibility calculations. The finalized model and its vectorizer were exported into production-ready files and integrated into the Flask-based back end, enabling real-time classification during user interactions.

4.2 Use Cases

The model supports several core use cases within the Faktz platform. The primary use case involves analyzing a TikTok transcript immediately after a user submits it. The text is received by the API, cleaned using the internal preprocessing pipeline, transformed into a TF-IDF vector, and classified into one of the three categories. The system also returns an associated confidence score and, when relevant, identifies organizational references appearing in supported claims.

Another key use case is providing structured feedback to users in a simple and interpretable format. Instead of exposing technical model outputs, the system translates predictions into credibility indicators and structured summaries. These cases collectively support fact-checking, misinformation awareness, and content evaluation, which form the core purpose of the Faktz platform.

4.3 Available Features

The classification component provides multiple functional features that enhance the overall behavior of the system. It performs automated text preprocessing, including contraction expansion, removal of links and emojis, character filtering, and lemmatization. This ensures that all text entering the model follows a consistent structure, reducing noise and improving prediction reliability. The real-time classification capability enables fast response times suitable for user-facing applications. The system also generates confidence percentages for each prediction, improving transparency and helping users interpret the level of certainty. Additionally, a resource extraction module identifies organizational names in supported claims using spaCy’s Named Entity Recognition. This feature adds contextual value by presenting recognizable sources directly to the user.

4.4 Strength and Uniqueness

The strengths of the model come from the synergy between structured data preprocessing, carefully curated training data, and a well-performing traditional machine learning pipeline:

- Strong performance achieved through structured preprocessing, balanced datasets, and appropriate model selection.
- TF-IDF provides effective linguistic sensitivity without the computational cost associated with deep learning approaches.
- Linear SVM enables rapid prediction times suitable for real-time and interactive fact-checking scenarios.
- The end-to-end analysis pipeline returns results in approximately five seconds, allowing users to receive instant feedback without noticeable delay.
- The calibration layer supports probability-based scoring, improving the clarity and reliability of credibility assessments.
- The resource extraction component enhances contextual understanding by identifying referenced organizations when a transcript contains a Supported Claim.
- The integration of classification accuracy, interpretability, fast processing, and contextual enrichment creates a unique user experience that differs from typical transcript classifiers on similar platforms.

4.5 Summary

The overall design of the model brings together a clear and practical approach to analyzing TikTok video transcripts. The model operates within a well-structured machine learning pipeline that prepares each piece of text carefully before classification, which helps the system handle the wide range of language styles typically found on social media. With the combination of TF-IDF features and a calibrated Linear SVM, the system is able to classify transcripts quickly and reliably, usually completing the entire process in about five seconds. This speed is important because it keeps the interaction smooth and makes the platform feel responsive and intuitive. The model's ability to provide confidence scores and automatically highlight referenced organizations adds meaningful context to the output, giving users a clearer picture of how credible the analyzed content might be. Taken together, these design choices give Faktz a unique blend of accuracy, transparency, and usability that supports its goal of helping users navigate online information more confidently.

Chapter 5: System Development

5.1 Overview

The development of the system followed a structured and sequential workflow that moved from data collection to model deployment. Since the goal of this system is to classify TikTok-style content into Opinion, Claim, and Supported Claim, the development process needed to account for the informal nature of online text and the inconsistencies that typically appear in user-generated content. Developing this capability requires more than training a single model. It involved constructing a full pipeline that could reliably handle messy, fast-paced social-media language while still producing results that users can understand and trust. For this reason, the work was divided into a series of interconnected notebooks, where each notebook represented a specific phase of the pipeline. This structure helped maintain clarity throughout the development journey and allowed each stage to be refined independently before progressing to the next.

The development began by compiling and validating the dataset to ensure that the collected material was suitable for training a classification model. This was followed by a detailed preprocessing and cleaning phase aimed at standardizing text that varied widely in style, tone, and structure. Once the text had been transformed into a consistent format, the next phase explored multiple modelling approaches and vectorization strategies to determine the most robust and efficient combination. The final stage focused on developing the production-ready classifier and preparing it for deployment through a Flask API.

Testing and evaluation are addressed separately in Chapter 6, allowing this chapter to focus entirely on the design and implementation.

Together, these phases form the foundation of the Faktz analytical engine, ensuring that the system is capable of handling real-world transcript input while maintaining speed, accuracy, and consistency throughout its operation.

5.2 System Development Tools and Configuration

The development of the system was supported by a collection of tools and libraries chosen specifically to handle large text datasets, perform extensive experimentation, and prepare the final model for deployment. Since the system needed to process informal and often noisy online content, the development workflow was built around Python due to its strong set of libraries for natural language processing and classical machine learning. All work was carried out in Google Colab, which made it easier to separate the project into multiple notebooks, with each notebook dedicated to a specific phase of the pipeline. This approach helped keep the entire workflow organized and made it easier to revisit or refine earlier stages without interrupting later work.

The project was intentionally split across several Python files, with each file representing one notebook and one stage in the development pipeline. The structure included:

1. **Synthetic Data Notebook - *DataGeneration_Templates.py***
    - Purpose: To generate additional examples for the Claim, Supported Claim, and Opinion categories, including TikTok-style sentences and template-based variations to increase dataset diversity.

    - Configuration & Tools:
        - Pandas for constructing and storing generated samples
        - Template-based text manipulation for producing structured synthetic data

This notebook expanded the dataset so the model could learn broader linguistic patterns.

2. **Merging Notebook - DataMerge.py**
    - Purpose: To merge the original TikTok dataset, the synthetic dataset, and a cleaned news dataset into one unified file, with column names and formats aligned across all sources.

    - Configuration & Tools:
        - Pandas for merging, reformatting, and deduplication
        - Regex for additional text normalization

This notebook ensured that all datasets matched structurally before inspection and modelling.

3. **Inspection Notebook - Inspection_2.py**
    - Purpose: To understand the structure and quality of the collected datasets before any modelling took place. The notebook examined class distribution, duplicate rows, text lengths, stopword ratios, and general noise levels.

    - Configuration & Tools:
        - Pandas for loading and examining datasets
        - Matplotlib & Seaborn for visualizing distributions
        - NumPy for statistical summaries

    This notebook served as the analytical foundation for later preprocessing decisions.

4. **Cleaning Notebook - *Preprocessing_&_Cleaning.py***
    - Purpose: To prepare all text for machine-learning tasks by applying a multi-stage cleaning pipeline. This notebook produced the final cleaned dataset used for model training.

    - Configuration & Tools:
        - ftfy and contractions for correcting text irregularities
        - Regex (re module) for removing URLs, emojis, and punctuation
        - NLTK for tokenization, stopword resources, and lemmatization fallback
        - spaCy (en_core_web_sm) for advanced lemmatization

    This notebook exported the cleaned dataset in a structured format suitable for modelling.

5. **Model-Selection Notebook (Classical Models) - Model_Selection_1.py**
    - Purpose: To compare traditional vectorization and classification methods, establishing baseline results before moving to advanced embedding approaches.

    - Configuration & Tools:
        - Scikit-Learn Vectorisers:
        - CountVectorizer (Bag-of-Words)
        - TfidfVectorizer (TF-IDF)
    - Scikit-Learn Models:
        - Logistic Regression
        - Random Forest
        - Linear SVM
    - Classification metrics: Accuracy, precision, recall, F1-score

    This notebook identified TF-IDF and Linear SVM as strong candidates.

6. **Embedding-Based Model Notebook - *Selection_2_WE_(2).py***
    - Purpose: To experiment with semantic vectorization methods and evaluate their performance on short, noisy text representing TikTok transcripts.

    - Configuration & Tools:
        - Gensim for loading large pre-trained embeddings:
            - Word2Vec (Google News)
            - FastText (Wiki-News Subwords)
        - NumPy for creating averaged sentence vectors
        - Scikit-Learn for SVM training and evaluation

    This notebook provided insight into how embedding models behave compared to TF-IDF.

7. **Final Model Development Notebook - Development.py**
    - Purpose: To train the final version of the model using TF-IDF and a calibrated Linear SVM, generate confidence scores, calculate credibility percentages, and export model artefacts.

    - Configuration & Tools:
        - TfidfVectorizer (with bigrams, sublinear TF, stopword removal)
        - LinearSVC + CalibratedClassifierCV
        - Joblib for saving the trained model and vectoriser

    This notebook produced the final deployable model used in the system.

8. **Deployment Notebook - *app.py* (Flask API)**
    - Purpose: To wrap the trained model into an API that performs classification in real time. The API handles internal cleaning, vectorization, prediction, confidence output, and resource extraction.

    - Configuration & Tools:
        - Flask for hosting the API endpoint
        - Joblib for loading model artefacts
        - spaCy for organization-name extraction
        - Regex + NLTK for internal cleaning functions

    This notebook transformed the model into a functional system component ready for integration into the final platform.

5.3 Uniqueness & Requirements of the System

The system stands out because it was developed as a complete analytical pipeline, not a single model. Its design responds directly to the limitations identified in the literature review, where most existing studies on online misinformation or claim classification either rely on a single data source, focus on one modelling technique, or limit their scope to binary outcomes.

5.3.1 Development Approach

1. Multi-Source Dataset Construction

The pipeline began with building a dataset that could represent both the informal, short, and inconsistent nature of TikTok content and the structured, factual form of news sources.

This required combining several data origins:

- Kaggle credibility datasets
- Synthetic TikTok-style samples generated using template-based methods
- Manually constructed Supported Claim examples
- A cleaned and standardized news dataset (merged using DataMerge.py)

This hybrid dataset created a linguistic diversity that does not appear in prior credibility-classification studies, which usually rely on a single homogeneous dataset.

2. Unified Dataset Formatting

Once collected, all datasets were formatted into a single, consistent structure in DataMerge.py. This alignment ensured that:

- formal news writing
- informal TikTok-style phrasing
- template-generated samples

all could be merged without losing semantic meaning or label consistency.

This cross-domain harmonization is rarely addressed in literature, where datasets from different sources are typically treated separately.

	claim_status	video_transcription_text
0	opinion	my colleagues are willing to say that one-thir...
1	Supported claim	A report from Reuters revealed that WASHINGTON...
2	Supported claim	Reuters said that WASHINGTON U.S. House of Rep...
3	Supported claim	US Census Bureau confirmed that found that soc...
4	claim	which abuts the Sahara to the north and has be...
...	...	...
47271	opinion	my sentiment is that a blue whale's heartbeat ...
47272	claim	The United States informed Germany shortly bef...

```
#  column order
df_main = df_main[["claim_status", "video_transcription_text"]]
df_news = df_news[["claim_status", "video_transcription_text"]]

df_merged = pd.concat([df_main, df_news], ignore_index=True)
df_merged = df_merged.sample(frac=1, random_state=42).reset_index(drop=True)
```

Figure 15: Merged Datasets Sample

3. Deep Multi-Stage Preprocessing Pipeline

To prepare the dataset for modelling, the system applied an extensive cleaning and normalization process. The pipeline included:

a. Unicode correction
b. Contraction expansion
c. Emoji, symbol, and URL removal
d. Regex-based noise filtering
e. spaCy lemmatization

This six-layer cleaning procedure was specifically engineered for noisy TikTok transcripts, which contain slang, emojis, inconsistent casing, and fragmented grammar. None of the reviewed studies used such a comprehensive and tailored preprocessing workflow.

4. Comparative Modelling Across Representation Families

To identify the most suitable approach for short, informal text, the system evaluated four major representation methods:

- Bag-of-Words
- TF-IDF
- Word2Vec embeddings
- FastText embeddings

Most existing research only evaluates one representation method. This project evaluated four, across two modelling paradigms, making the comparison far more extensive than what is typically reported.

5. Evidence-Based Model Selection

After extensive testing, the TF-IDF + Linear SVM combination consistently outperformed the alternatives in:

- Accuracy
- F1-score
- Generalization ability
- Inference speed

The choice of final model was therefore based on empirical evidence, not assumptions. This aligns with best practices noted in the literature but goes beyond them by evaluating a wider range of methods.

6. Probability Calibration and Credibility Scoring

Most SVM-based systems produce only hard labels. This project extends this by applying probability calibration using CalibratedClassifierCV, which enabled:

- probability-based output
- interpretable confidence scores
- credibility scoring

This calibration layer significantly increases transparency and interpretability, a feature not commonly present in related academic systems.

7. Integrated Source Extraction

For any input classified as a Supported Claim, the system activates an additional layer of analysis implemented in app.py, where referenced entities are extracted using spaCy NER. The final output therefore contains:

- The classification label
- The confidence score
- Any extracted organizations

No reviewed study integrates source extraction directly into the credibility-classification pipeline.

8. Real-Time Integration With the Faktz Website

The model is fully integrated into the Faktz website through a Flask API. When the user submits text on the platform, the website sends it directly to the backend, where the complete pipeline runs automatically. The API:

- cleans and normalizes the text,
- applies the saved TF-IDF vectorizer,
- predicts the credibility category using the calibrated Linear SVM,
- generates confidence and credibility scores
- extracts referenced organizations if the text is a Supported Claim.

The processed results are then returned to the website in a structured JSON response. This entire end-to-end flow from user input to final output completes in about five seconds, giving Faktz real-time credibility analysis.

5.3.2 Code Development

1. Synthetic Data generation

The first step involved creating synthetic TikTok-style statements to expand the dataset and strengthen class balance. The code used a modular template-based approach, which allowed the system to generate informal, conversational examples that mirror the linguistic patterns observed on TikTok.

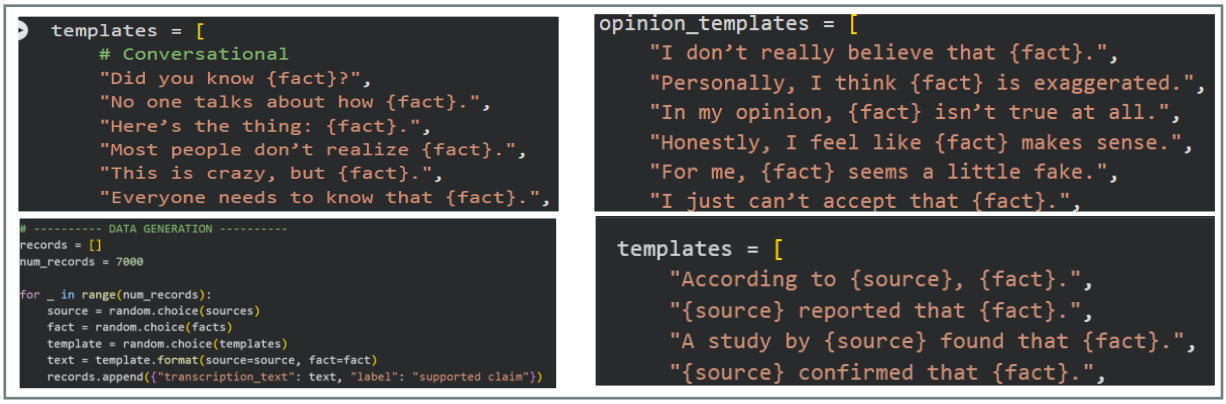


Figure 16: Data Generation templates Example

These templates contained placeholders such as {fact} and {source}, allowing the system to automatically create realistic variations of Claims, Opinions, and Supported Claims. This approach enabled the dataset to capture both the linguistic diversity and the informal style of real online conversations, which traditional datasets do not provide.

Using these templates, the code randomly selected facts, sources, and sentence structures to generate thousands of unique statements. Each template was combined programmatically with different elements, ensuring that every generated sample sounded natural while still maintaining the label's meaning. Duplicate rows were removed to preserve quality, and the final output was a rich, balanced, and varied synthetic dataset that significantly strengthened the training process. This randomised template-driven approach created data that could not be found in existing datasets and played a crucial role in improving the model's ability to understand and classify short, informal user-generated text.

```
print(df.duplicated().sum())
df=df.drop_duplicates()

565

df = pd.DataFrame(records)
df.to_excel("opinion_dataset.xlsx", index=False)
df.to_csv("opinion_dataset.csv", index=False)
```

2. Merging the Data

In the data merging stage, the goal was to bring together all data sources into a single, well-structured and balanced dataset that could be used for training. The script first loads two main inputs. The file final_dataset(csv).csv contains the TikTok style and synthetic data created earlier, while True.csv contains factual news articles that will be transformed into claims and supported claims.

The news data is cleaned by removing irrelevant columns such as subject, date, and title, keeping only the raw text. A generic clean_text function then standardizes encoding problems and extra whitespace to ensure the text field is consistent. After this, two specialized cleaning functions are applied. clean_claim_text removes explicit source names such as “Reuters” or “BBC” and introductory reporting phrases from half of the rows, so that these sentences become standalone claims. clean_supported_text keeps the overall structure of the sentence but trims leading metadata from the remaining rows, which are used as supported claims. The news data frame is shuffled, split into two halves, cleaned using these functions, and then labelled as claim and Supported claim respectively before being recombined into the final dataset and shuffled again.

	title	text	subject	date
0	As U.S. budget fight looms, Republicans flip t...	WASHINGTON (Reuters) - The head of a conservat...	politicsNews	December 31, 2017
1	U.S. military to accept transgender recruits o...	WASHINGTON (Reuters) - Transgender people will...	politicsNews	December 29, 2017
2	Senior U.S. Republican senator: 'Let Mr. Muell...	WASHINGTON (Reuters) - The special counsel inv...	politicsNews	December 31, 2017
3	FBI Russia probe helped by Australian diplomat...	WASHINGTON (Reuters) - Trump campaign adviser ...	politicsNews	December 30, 2017
4	Trump wants Postal Service to charge 'much mor...	SEATTLE/WASHINGTON (Reuters) - President Donal...	politicsNews	December 29, 2017

Figure 18: Original News Dataset

```python
def clean_text(t):
    t = str(t)
    t = t.encode("utf-8", "ignore").decode("utf-8", "ignore")
    t = re.sub(r"Â", "", t)
    t = re.sub(r"\s+", " ", t).strip()
    return t

df_news["text"] = df_news["text"].apply(clean_text)
```

```python
def clean_supported_text(text):
    text = re.sub(r"^[A-Z\s/]+?\(\)\s*-\s*", "", text, flags=re.I)
    text = re.sub(r"\s+", " ", text).strip()
    return text
```

```python
def clean_claim_text(text):
    text = re.sub(r"\b(Reuters|Associated Press|AP|BBC|CNN|Bloomberg|AFP)\b", "", text, flags=re.I)
    text = re.sub(r"^[A-Z\s/]+?\(\)\s*-\s*", "", text, flags=re.I)
    text = re.sub(r"^.*?-\s*", "", text, flags=re.I)
    text = re.sub(r"\s+", " ", text).strip()
    return text
```

Figure 17: News Cleaning Functions

```python
df_news = df_news.sample(frac=1, random_state=42).reset_index(drop=True)
split_index = len(df_news) // 2

df_claims = df_news.iloc[:split_index].copy()
df_supported = df_news.iloc[split_index:].copy()

df_claims["text"] = df_claims["text"].apply(clean_claim_text)
df_supported["text"] = df_supported["text"].apply(clean_supported_text)

df_claims["label"] = "claim"
df_supported["label"] = "Supported claim"

df_final = pd.concat([df_claims, df_supported], ignore_index=True)
df_final = df_final.sample(frac=1, random_state=42).reset_index(drop=True)
```

To integrate the news data with the TikTok style dataset I renamed the columns to match the main schema, setting video_transcription_text as the text field and claim_status as the label field. A further transformation step then makes the supported claims sound more conversational and closer to TikTok phrasing. A list of informal reference patterns such as “According to {}” and “Based on a post from {}” is defined and the function make_source_casual searches each sentence for news outlet names like “Reuters” or “CNN” . When a match is found the source is wrapped inside one of these casual templates and reattached to the cleaned text. This creates supported claim sentences that still retain their factual origin but read in a more natural, social media like way. The resulting news dataframe is then merged with the main generated dataset (df_main) which has its own columns reduced to the same two key fields. Both dataframes are concatenated, shuffled, and saved as data_merged.csv. At this point the project has a single unified dataset that combines synthetic TikTok style statements with rephrased real world news based claims and supported claims.

```
def make_source_casual(text):
    match = re.search(r"\b(Reuters|BBC|CNN|Bloomberg|AFP|Associated Press|AP)\b", text, flags=re.I)
    if match:
        source = match.group(0)
        phrase = random.choice(casual_refs).format(source)
        text = re.sub(r"\(?\b" + source + r"\b\)?\s*-\s*", "", text, flags=re.I)
        text = phrase + " " + text.strip()
    return text.strip()

df_final.loc[df_final["claim_status"] == "Supported claim", "video_transcription_text"] = (
    df_final.loc[df_final["claim_status"] == "Supported claim", "video_transcription_text"]
    .apply(make_source_casual)
)
```

Figure 19: Code snippet of the function

The second part focuses on topic based balancing to avoid the model overfitting to dominant themes. The merged dataset is loaded again and preprocessed to remove blanks. A TF IDF vectorizer is then applied to encode all texts numerically and Keans clustering is used to detect approximately ten latent topic clusters, which are stored in a topic_cluster column . For interpretability the script prints the top keywords for each cluster so that rough topic themes can be inferred. To further control topic distribution a manual mapping of topic labels such as “health”, “beauty”, “fitness”, “sports”, “tech”, “finance”, “entertainment”, and “environment” is defined along with a set of ready made example sentences for each topic and label combination. Each existing row is randomly assigned one of these topic labels, and the script checks how many samples each topic contains. Whenever a topic has fewer rows than the target threshold, additional synthetic examples are drawn from the corresponding topic_templates list and appended as new rows with the appropriate claim_status, video_transcription_text, and topic metadata. The final balanced dataframe is shuffled again and saved as final_dataset_balanced.csv.

This step ensures that the training data is not only merged and cleaned but also evenly distributed across different real-world themes, which strengthens the robustness and generalization ability of the final model.

```
 Top keywords per detected topic:
Topic 1: sounds, real, dolphins, mammals, think, highkey, honest, imo, correct, fake
Topic 2: friends, willing, view, say, world, earth, convinced, understanding, impression, understand
Topic 3: trump, said, republican, house, president, clinton, white, senate, campaign, donald
Topic 4: family, willing, wager, bet, view, world, colleagues, understands, say, impression
Topic 5: said, government, state, reuters, president, people, year, party, minister, united
Topic 6: colleagues, earth, world, opinion, believe, view, moon, feel, don, humans
Topic 7: reported, confirmed, stated, highlighted, global, university, research, emphasized, study, accelerating
Topic 8: korea, north, korean, nuclear, said, china, south, missile, trump, pyongyang
Topic 9: discovered, read, claim, colleague, friend, claiming, mentioning, news, tv, media
Topic 10: learned, colleague, claim, media, website, discussion, news, board, internet, social
```

3. Inspection

After merging all datasets into a single file, the next stage involved performing a thorough inspection to understand the structure, quality, and distribution of the data before applying any preprocessing. This step was essential because the merged dataset combined formal news text, synthetic TikTok-style statements, and conversational examples generated using templates. Ensuring that these diverse sources were clean, consistent, and balanced was crucial for the reliability of the later modelling phase. The inspection involved analytical, statistical, and visual examination of the dataset.

The dataset was first loaded and examined to verify its overall structure, including shape, column consistency, data types, and the completeness of the final schema (video_transcription_text, claim_status). A review of the class distribution was then performed, supported by a bar-chart visualization, to identify any imbalance across the Opinion, Claim, and Supported Claim categories. Understanding this distribution early on was important, as imbalanced classes can negatively influence model performance and would later guide any balancing strategies.

Checks for missing values and repeated text entries were then carried out. Duplicate statements were removed to prevent information leakage and inflated performance during model evaluation. Additional exploratory metrics such as text length and word count were analyzed to characterize the linguistic nature of the dataset. These analyses confirmed that the dataset predominantly consisted of short statements, a feature typical of TikTok-style content, making frequency-based representations such as TF-IDF an appropriate modelling choice.

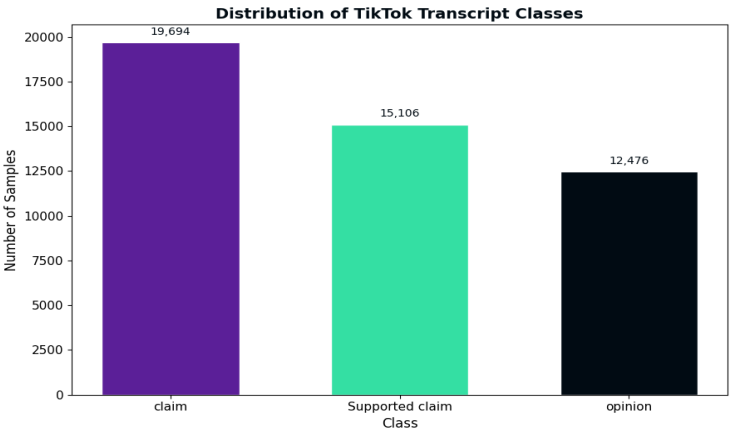
To gain further insight into each label category, stopword proportions were calculated and the most frequent terms for each class were extracted. These observations highlighted meaningful linguistic differences between the three credibility categories. Word-cloud visualizations were also generated using the project's color palette, providing a clear, intuitive illustration of dominant vocabulary patterns within each class.

Overall, this inspection stage established a thorough understanding of the merged dataset and confirmed its readiness for the subsequent cleaning, preprocessing, and modelling phases. The insights gained here directly supported method selection and strengthened the reliability of the later stages of development.

```
Dataset Shape: (47276, 2)
================================================

Column Names: ['claim_status', 'video_transcription_text']
================================================

Basic Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 47276 entries, 0 to 47275
Data columns (total 2 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   claim_status            47276 non-null  object
 1   video_transcription_text  47275 non-null  object
dtypes: object(2)
memory usage: 738.8+ KB
None
```




```
Transcript length stats:
count    44006.000000
mean      1121.607031
std       1583.191279
min          3.000000
25%         87.000000
50%        130.000000
75%       1923.000000
max      29937.000000
Name: text_length, dtype: float64
```

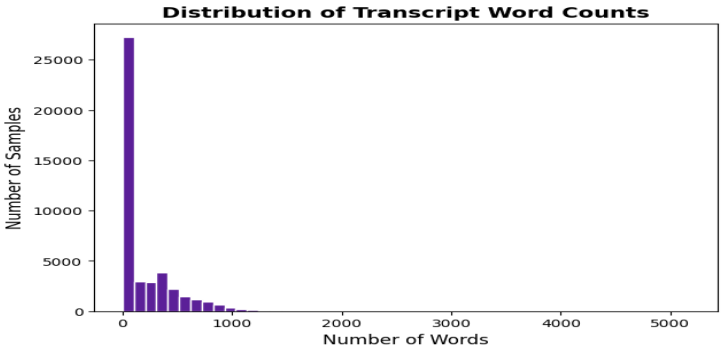


Figure 20: Inspection Phase Results

4. Preprocessing and Cleaning

Once the dataset had been inspected and validated, the next stage involved preparing the text for machine learning through a multi-layer preprocessing pipeline. This phase played a critical role in enhancing data quality, removing noise, and ensuring that all statements whether synthetic, news-generated, or TikTok-style were transformed into a uniform linguistic format. The preprocessing process followed a structured sequence of four main phases:

4.1 Phase 1: Correction of Text Irregularities

The first phase focused on correcting encoding issues and fixing broken text patterns often found in user-generated statements. The function fix_text from the *ftfy* library was applied to each sample to restore any corrupted characters and ensure that all text fields were consistently readable. This step was particularly important because the merged dataset included content produced by templates, manual text entry, and scraped factual sentences, all of which may contain irregular spacing or encoding fragments.

4.2 Phase 2: Expansion of Contractions

The next phase expanded contractions (such as “don’t”, “isn’t”, or “I’ve”) into their full forms using the *contractions* library. This standardization step reduced linguistic ambiguity and ensured that the model processed words consistently. For example, “don’t believe this” and “do not believe this” become equivalent after expansion, which improves downstream vectorization and reduces noise in the vocabulary.

```
df["phase2_contractions"] = df["phase1_fixed"].apply(lambda x: contractions.fix(str(x)))
```

4.3 Phase 3: Removal of URLs, Special Characters, and Noise

A custom cleaning function was then used to remove URLs, symbols, emojis, and any non-alphabetical characters. All text was lowercased to standardize formatting. This step ensured that only meaningful linguistic content remained for analysis and prevented irrelevant characters from influencing TF-IDF representations.

```
def clean_special_chars(text):
    text = text.lower()
    text = re.sub(r"http\S+|www\S+|https\S+", "", text)  # remove urls
    text = re.sub(r"[^a-z\s]", " ", text)               # remove non-letters
    return text.strip()

df["phase3_cleaned"] = df["phase2_contractions"].apply(clean_special_chars)
```

4.4 Phase 4: Lemmatization

The final preprocessing phase involved applying spaCy’s lemmatizer to convert each token to its root form (e.g., “running” → “run”, “studies” → “study”). Lemmatization reduces the vocabulary size and improves the model’s ability to recognize semantically related words. This was especially valuable for short statements, where every token carries high weight.

```
def spacy_process(text):
    doc = nlp(text)
    tokens = [token.lemma_ for token in doc]
    return {"lemmas": " ".join(tokens)}

df["phase4_spacy"] = df["phase3_cleaned"].apply(spacy_process)
```

4.5 Exporting

After completing all preprocessing phases, the relevant columns were extracted and saved as a clean dataset ready for vectorization and modelling:

- **claim_status** – target label
- **phase3_cleaned** – cleaned baseline text
- **lemmas** – lemmatized text used for training

Duplicates were removed once more to ensure that no repeated entries remained.

5. Model Selection

After preprocessing, the next stage involved identifying the most suitable representation and classification method for short, informal TikTok-style statements. This stage was critical because the dataset contained conversational language, mixed formality levels, and highly variable sentence structures. Two dedicated notebooks were used to explore a wide range of approaches: one focusing on traditional vectorization and classical machine-learning models, and the other evaluating semantic embedding techniques. This comprehensive comparison ensured that the final model choice was based on empirical performance rather than theoretical preference.

5.1 Classical Machine-Learning Models and Traditional Vectorizers

The model-selection stage begins by loading the cleaned dataset and preparing the target labels for machine-learning algorithms. The text data is extracted from the lemmatized column, and the credibility labels are numerically encoded to ensure compatibility with scikit-learn classifiers. The dataset is then divided into training, validation, and testing sets using a 70–15–15 structure. Splitting in two steps avoids information leakage and ensures that the validation and test partitions remain fully independent.

```
print(f"Train: {len(X_train)} samples")
print(f"Validation: {len(X_val)} samples")
print(f"Test: {len(X_test)} samples")
```

```
Train: 30835 samples
Validation: 6607 samples
Test: 6608 samples
```

5.1.1 Defining Classical Models and Evaluation Logic

A collection of classical machine-learning models is created, including Logistic Regression, Linear SVM, and Random Forest. Each model is initialized with balanced class weights to counteract variations in class frequency. To ensure consistent comparison across models, an evaluation function is defined. This function fits a model, makes predictions, and calculates key metrics such as precision, recall, F1-score, and accuracy. Training time is also captured to assess computational efficiency.

```
models = {
    "Logistic Regression": LogisticRegression(max_iter=1000, class_weight='balanced', random_state=42),
    "Linear SVM": LinearSVC(class_weight='balanced', random_state=42),
    "Random Forest": RandomForestClassifier(n_estimators=200, class_weight='balanced', random_state=42)
}
```

```
def evaluate_model(model, X_train_vec, y_train, X_val_vec, y_val):
    start = time.time()
    model.fit(X_train_vec, y_train)
    end = time.time()
    train_time = end - start

    y_train_pred = model.predict(X_train_vec)
    y_val_pred = model.predict(X_val_vec)

    train_acc = accuracy_score(y_train, y_train_pred)
    val_acc = accuracy_score(y_val, y_val_pred)

    precision, recall, f1, _ = precision_recall_fscore_support(y_val, y_val_pred, average='weighted', zero_division=0)
    return round(train_time,2), train_acc, val_acc, precision, recall, f1
```

5.1.2 Bag-of-Words Representation

The first representation tested is Bag-of-Words using a CountVectorizer with up to 1,000 features and bigram support. The vectorizer is fitted on the training text and then applied to the validation set. Each classical model is trained and evaluated using these Bag-of-Words vectors. This step establishes a baseline for comparison with more expressive methods.

```
# BOW
bow_vectorizer = CountVectorizer(max_features=1000, stop_words='english', ngram_range=(1,2))
X_train_bow = bow_vectorizer.fit_transform(X_train)
X_val_bow = bow_vectorizer.transform(X_val)
```

5.1.3 TF-IDF Representation with Varying Feature Sizes

Next, TF-IDF vectorization is explored using two different feature limits: 50 and 1,000. For each feature size, the vectorizer is trained on the training data and applied to the validation data. The same set of classical models is then evaluated. TF-IDF typically performs better on short text because it emphasizes distinctive words while down-weighting common conversational terms, which proved useful for the TikTok-style dataset.

```
max_feature_values = [50, 1000]

for mf in max_feature_values:
    tfidf = TfidfVectorizer(max_features=mf, ngram_range=(1,2), stop_words='english')
    X_train_tfidf = tfidf.fit_transform(X_train)
    X_val_tfidf = tfidf.transform(X_val)
```

5.1.4 Comparative Results

The comparison indicates that both Bow and TF-IDF can achieve high performance when using 1,000 features, but TF-IDF produces more consistent results across models. Although Random Forest reaches strong accuracy and F1-scores, its training time is significantly higher than the other models, making it inefficient for real-time use. Logistic Regression remains stable but slightly weaker across all metrics.

When evaluating all metrics together the TF-IDF Linear SVM configuration offers the strongest overall balance. It achieves high performance while maintaining extremely fast training. This combination therefore represents the most efficient and reliable option for short TikTok-style credibility classification.

All Results Summary:

	Vectorizer	Max Features	Model	Train Time (s)	Train Accuracy	Val Accuracy	Precision	Recall	F1
0	BoW	1000	Logistic Regression	5.76	0.968	0.954	0.956	0.954	0.954
1	BoW	1000	Linear SVM	17.06	0.971	0.957	0.958	0.957	0.957
2	BoW	1000	Random Forest	115.00	0.977	0.965	0.966	0.965	0.965
3	TF-IDF	50	Logistic Regression	0.54	0.706	0.709	0.774	0.709	0.709
4	TF-IDF	50	Linear SVM	0.37	0.714	0.716	0.775	0.716	0.720
5	TF-IDF	50	Random Forest	41.50	0.783	0.737	0.804	0.737	0.741
6	TF-IDF	1000	Logistic Regression	1.20	0.956	0.954	0.955	0.954	0.954
7	TF-IDF	1000	Linear SVM	1.01	0.967	0.962	0.963	0.962	0.962
8	TF-IDF	1000	Random Forest	118.91	0.977	0.967	0.968	0.967	0.967

Figure 21: Model Selection Training Results

5.1.5 Learning-Curve Analysis

After training the models SVM proved to have the best results, so to understand how it behaves as training data increases, the code generates learning curves for:

- Bag-of-Words + SVM
- TF-IDF + SVM

Learning curves reveal the model's generalization behavior and show whether performance gains plateau or improve with additional data. This analysis helps confirm that TF-IDF is more stable and generalizes better for short statements.

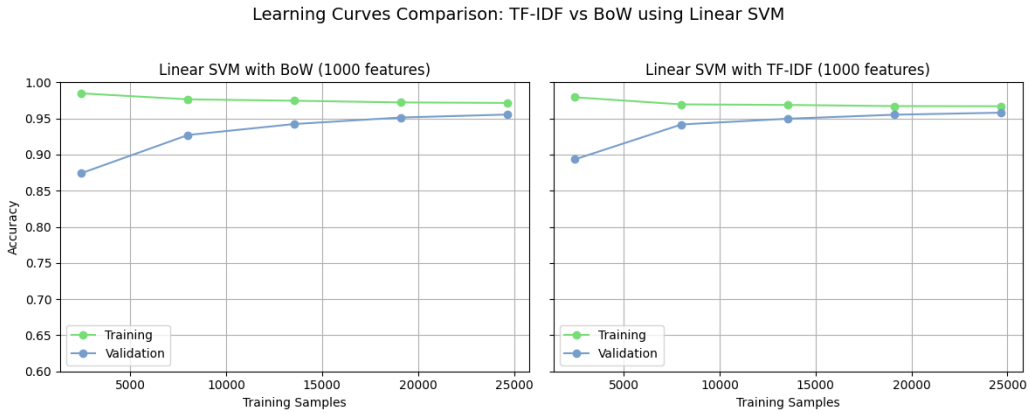


Figure 22: Learning Curve

5.1.6 Increasing TF-IDF Feature Capacity

Based on earlier observations, TF-IDF is re-evaluated using a larger vocabulary of 2,000 features. This increases expressiveness and allows the model to capture more linguistic variety. The updated vectorizer is fitted on the training data and transforms both training and testing text for further evaluations.

A check is also performed to ensure that no overlapping text exists between the training and validation partitions, preventing data leakage.

```
Train shape: (30835, 2000)
Validation shape: (6607, 1000)
Number of overlapping samples: 0
```

```
TfidfVectorizer
TfidfVectorizer(max_features=2000, ngram_range=(1, 2))
```

5.1.7 Hyperparameter Tuning for Linear SVM

To refine the best-performing model, a parameter grid is defined for Linear SVM:

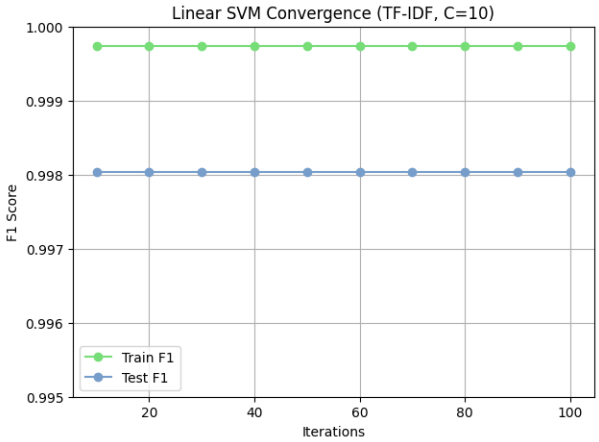
- regularization strength (C)
- loss function (hinge vs. squared hinge)
- tolerance values
- iteration limits

GridSearchCV iterates across these combinations and selects the configuration that yields the highest weighted F1 score. This ensures that the final classifier is optimized for both performance and class balance.

```
Fitting 5 folds for each of 90 candidates, totalling 450 fits
Best Parameters: {'C': 10, 'loss': 'squared_hinge', 'max_iter': 10, 'tol': 0.0001}
Best F1 Score: 0.9966594687566372
```

5.1.8 Iterative Model-Training Analysis

The behavior of the Linear SVM is further examined through an incremental training loop. The maximum iteration value is progressively increased in small steps, and after each step, the model is trained and evaluated. This reveals how quickly the model converges and whether performance stabilizes or continues to improve.



The incremental-iteration analysis shows that the Linear SVM reaches optimal performance almost immediately, the model maintains a constant Train & Test F1-score, with no meaningful improvement or decline as iterations increase. This indicates that the classifier converges extremely quickly under the TF-IDF representation and does not benefit from additional optimization cycles.

The plotted convergence curve further confirms this behavior: both the training and testing F1-scores appear as flat, stable lines, showing no signs of overfitting, underfitting, or instability. Such consistency across all iteration steps demonstrates that Linear SVM not only performs strongly but also maintains its generalization ability regardless of training duration.

Conclusion Based on the Iterative Results

Because the Linear SVM produced high and stable performance across all metrics, converged quickly, and demonstrated strong generalization even at low iteration counts, it was selected as the final model for development and calibration. This reliability, combined with its efficiency and robustness on short TikTok-style text, made it the most suitable choice for deployment in the Faktz pipeline.

5.2 Alternative Word-Representation Techniques

To further validate the selection of TF-IDF, additional experiments were conducted using semantic word-embedding techniques. The code in the embedding-evaluation module loads two pre-trained vector models: Word2Vec (Google News) and FastText (Wiki-News) and prepares them for sentence-level representation. Since these pre-trained models operate at the token level, the code first tokenizes each input statement and retrieves the corresponding word vectors. When a token does not exist in the embedding vocabulary, it is skipped. The remaining word vectors in each statement are then averaged to form a single 300-dimensional sentence embedding.

```
Generating Word2Vec sentence embeddings...
Train Embeddings: 100%|██████████| 30835/30835 [00:08<00:00, 3782.41it/s]
Test Embeddings: 100%|██████████| 13216/13216 [00:03<00:00, 3902.32it/s]
Embeddings generated in 11.59 seconds
Train embedding shape: (30835, 300)
Test embedding shape: (13216, 300)
```

```
Generating FastText sentence embeddings...
Train Embeddings: 100%|██████████| 30835/30835 [00:05<00:00, 5400.95it/s]
Test Embeddings: 100%|██████████| 13216/13216 [00:02<00:00, 5467.18it/s]Embeddings generated in 8.18 seconds
Train Vector Shape: (30835, 300)
Test Vector Shape:  (13216, 300)
```

These sentence embeddings are subsequently fed into a Linear SVM classifier using the same train/validation/test splits as in the TF-IDF experiments. The evaluation block computes accuracy, precision, recall, and F1-score to maintain a consistent comparison across all models.

```
start_train = time.time()

svm_clf_fasttext = SVC(kernel='linear', class_weight='balanced', random_state=42)
svm_clf_fasttext.fit(X_train_vecs, y_train)

end_train = time.time()
training_time = round(end_train - start_train, 2)
print(f"Training completed in {training_time} seconds.")


Training Support Vector Machine (SVM) classifier...
Training completed in 89.02 seconds.
```

```
svm_clf_word2vec = SVC(kernel='linear', class_weight='balanced', random_state=42)
svm_clf_word2vec.fit(X_train_vecs, y_train)

end_train = time.time()
training_time = round(end_train - start_train, 2)
print(f"Training completed in {training_time} seconds.")


Training Support Vector Machine (SVM) classifier...
Training completed in 68.84 seconds.
```

During experimentation, several limitations became clear. Because TikTok statements are short and often include emerging slang, casual phrasing, and creative spellings, many tokens failed to match the vocabularies of the pre-trained embedding models. As a result, some statements were represented with partially missing information or, in rare cases, reduced to zero-vectors.

Furthermore, the averaging procedure reduced sentence distinctiveness: statements with different meanings but similar token embeddings collapsed into similar vector regions, making it difficult for the classifier to separate Claim, Opinion, and Supported Claim categories effectively.

These limitations were reflected in the evaluation metrics, where both Word2Vec and FastText achieved noticeably lower scores than TF-IDF models. When all factors (accuracy, representation fidelity, vocabulary coverage, and computational cost) were considered together, the embedding-based approaches performed less reliably than TF-IDF, further reinforcing the selection of TF-IDF + Linear SVM as the most appropriate configuration for short, fast-paced social-media text.

	Embedding	Accuracy	Precision	Recall	F1-Score	Embedding Time (s)	Training Time (s)	Inference Time (s)
0	Word2Vec (GoogleNews)	0.838	0.846	0.838	0.838	{11.59}	68.84	20.95
1	FastText (Wiki-News via API)	0.757	0.833	0.757	0.743	8.18	89.02	27.42

6. Final Model Development

The final development stage transformed the selected configuration; TF-IDF vectorization combined with a calibrated Linear SVM classifier into a fully trained, evaluated, and deployable model. This stage ensured that the model achieved high accuracy, produced reliable confidence scores, and could operate in real time when integrated into the Faktz web system. The development workflow consisted of four main components: dataset preparation, vectorization, training and calibration, model evaluation, and finally exporting the trained components for deployment.

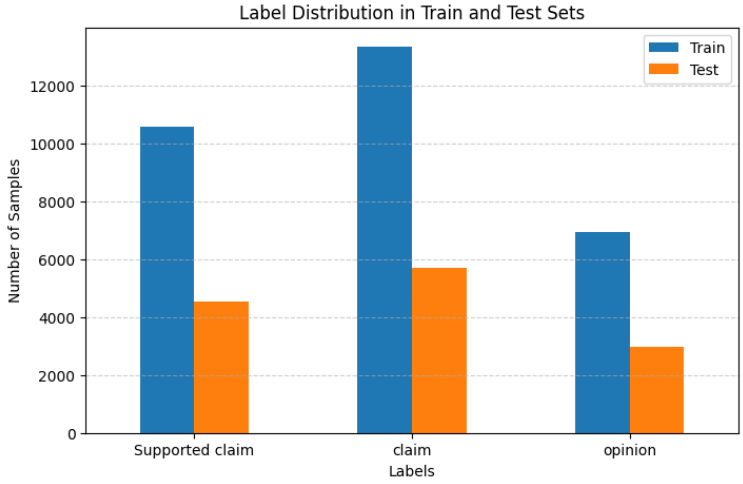
6.1 Data Preperation

The development process started by loading the preprocessed dataset and ensuring that all entries contained valid lemmatized text and labels. The data was then divided into training and testing subsets using a stratified split to preserve class balance. A visualization of class distribution was produced to confirm that each label was proportionally represented in both subsets.

```
X=data['lemmas']
y=data['claim_status']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42,stratify=y)

print("X size:", X_train.shape)
print("y size:", X_test.shape)
```

```
X size: (30835,)
y size: (13215,)
```



6.2 TF-IDF Vectorization

The vectorizer was configured to capture both unigrams and bigrams while reducing noisy or overly common terms. Several parameters were introduced to improve representation quality:

- sublinear_tf=True to scale term frequencies logarithmically
- ngram_range=(1,2) to capture short phrases
- max_df=0.9 and min_df=2 to remove overly common and extremely rare terms
- English stop words removal

```
tfidf = TfidfVectorizer(
    sublinear_tf=True,
    stop_words="english",
    ngram_range=(1,2),
    max_df=0.9,
    min_df=2
)
X_train_tfidf = tfidf.fit_transform(X_train)
X_test_tfidf  = tfidf.transform(X_test)
```

The vectorizer was fitted on the training text and applied to the test text, producing the sparse matrices required by the Linear SVM classifier.

6.3 Training and Calibration of the Linear SVM

A Linear SVM served as the base classifier. To enable probability estimation which LinearSVC does not provide natively, the model was wrapped inside a CalibratedClassifierCV.

Calibration allowed the model to output meaningful probability values for each class, enabling the system to provide:

- A predicted label
- A confidence score

The calibrated model was then trained on the TF-IDF vectors of the training set.

```
base_svm = LinearSVC(
    C=10,
    class_weight="balanced",
    random_state=42,
    dual="auto",
    max_iter=1000
)
```

```
calibrated_svm = CalibratedClassifierCV(base_svm, cv=5)
calibrated_svm.fit(X_train_tfidf, y_train)
```

CalibratedClassifierCV
estimator: LinearSVC
LinearSVC

6.7 Model Evaluation and Performance Analysis

Evaluation was conducted on both the training set and the held-out test set. Results included:

- overall accuracy
- precision, recall, and F1-score per label
- confusion matrix visualization

The model achieved high performance across all metrics, confirming its ability to distinguish between the three categories. The confusion matrix provided insight into class boundaries, particularly between Claim and Supported Claim, where linguistic differences can be subtle.

6.8 Confidence Scoring Computation

The equation computes both the predicted credibility label and the confidence score produced by the calibrated SVM classifier. First, the model generates a probability distribution for each statement, indicating how strongly it belongs to each class. The highest probability in this distribution is then extracted and multiplied by 100 to form a percentage-based confidence score, representing the model's certainty in its prediction.

At the same time, the index of this highest probability is used to retrieve the corresponding class label from the model's stored class list. Together, these operations translate raw probability outputs into an interpretable prediction consisting of a final label and a confidence value that can be presented clearly to end users.

```
proba = calibrated_svm.predict_proba(X_test_tfidf)
confidence = np.max(proba, axis=1) * 100
pred_labels = calibrated_svm.classes_[np.argmax(proba, axis=1)]
```

6.9 Exporting the Final Model and Vectorizer

After confirming the model's performance, the trained classifier and TF-IDF vectorizer were saved for deployment, this allowed the website backend to load the model instantly without retraining, ensuring real-time inference within approximately **5 seconds** from input to final output.

```
joblib.dump(calibrated_svm, "faktz_final_model.pkl")
joblib.dump(tfidf, "faktz_tfidf_vectorizer.pkl")
```

7. System Integration and API Deployment

To enable communication between the trained model and the Faktz website, a lightweight Flask-based API was developed . The API performs all necessary

7.1 Model and Vectorizer Loading

The API loads the trained TF-IDF vectorizer and calibrated SVM classifier into memory as soon as the server starts, this eliminates the need for retraining during runtime and enables instant access for prediction. Keeping both components preloaded ensures low latency and supports the system's five-second end-to-end response time.

```
model = joblib.load("faktz_final_model.pkl")
vectorizer = joblib.load("faktz_tfidf_vectorizer.pkl")
```

7.2 Receiving Input from the Website

The API exposes a /predict endpoint that receives text submitted from the Faktz web interface. User input arrives as raw, unprocessed text and is passed through the internal cleaning pipeline.

```
@app.route("/predict", methods=["POST"])
def predict():
    data = request.get_json()
    text = data.get("text", "")
```

7.3 The text undergoes a fast-cleaning pipeline that mirrors the preprocessing used during training, ensuring consistent formatting.

```
# Validate input
if not text or not isinstance(text, str):
    return jsonify({
        "success": False,
        "error": "Invalid or missing 'text' field."
    }), 400

# Clean text internally
cleaned = clean_and_lemmatize(text)
```

7.4 The cleaned text is converted into numerical vectors using the saved TF-IDF vectorizer.

```
X = vectorizer.transform([cleaned])
```

7.5 The calibrated SVM generates the predicted label and confidence score from the probability distribution.

```
# Transform input
proba = model.predict_proba(X)[0]
label = model.classes_[np.argmax(proba)]
confidence = np.max(proba) * 100
```

7.6 For Supported Claims, a source-extraction step uses spaCy to identify possible referenced organizations.

```
# Extract resource if Supported Claim
resource = extract_resource_name(text) if label.lower() == "supported claim" else None
```

7.7 The system returns a structured JSON response containing the prediction, confidence, and any extracted sources.

```
return jsonify({
    "label": label,
    "confidence (%)": round(confidence, 2),
    "resource": resource
}), 200
```

5.3.3 Uniqueness of the system summary:

1. **Pipeline-Level Uniqueness**
   - First pipeline (based on reviewed literature) to classify Opinion, Claim, and Supported Claim simultaneously.
   - Hybrid dataset combining TikTok-style synthetics, news data, and Kaggle sources.
   - Multi-stage cleaning pipeline tailored for noisy, short and long form transcripts.
   - Real-time predictions including classification, confidence, credibility scoring, and source extraction.

2. **Modelling Uniqueness**
   - Evaluation of four representation approaches: BoW, TF-IDF, Word2Vec, FastText.
   - Comparative analysis across classical ML and embedding-based models.
   - TF-IDF + Calibrated SVM selected based on empirical, not theoretical, justification.
   - Calibration-enabled probability outputs (rare in SVM-based systems).

3. **Functional Uniqueness**
   - Credibility percentage derived from calibrated confidence scores.
   - Extraction of referenced organizations using spaCy NER for Supported Claims.
   - End-to-end analysis achieved in ~5 seconds.

4. **Uniqueness Compared to Literature**
   - No prior study integrates dataset engineering, deep cleaning, representation comparison, model calibration, credibility scoring, and source extraction in one system.
   - Prior work typically focuses on binary outcomes (true/false), not three-way categorization.
   - TikTok-style transcripts as input remain largely unexplored in academic research.
   - Calibration + credibility scoring + organization extraction is entirely novel.

5.4 Summary

Chapter 5 brought together every stage of developing the system and showed how the ideas introduced earlier in the project were finally shaped into a working credibility-classification system. Since the system requirements and the uniqueness analysis were closely connected, they were merged into a single section to give a clearer picture of why the system was designed the way it was and what makes it different from existing approaches. The chapter first outlined how the dataset was built by combining Kaggle entries, template-generated TikTok-style statements, and carefully restructured news content. After merging these sources, the dataset went through a full inspection phase to understand its structure, balance, and linguistic behavior before any modelling took place.

The preprocessing stage created a clean and standardized version of the text by correcting irregularities, expanding contractions, removing noise, and applying lemmatization. This ensured that all statements entering the model shared a consistent style and format. A series of modelling experiments followed, exploring both traditional techniques and word-embedding approaches. The results showed that although embeddings like Word2Vec and FastText capture semantic relationships, they struggled with the short, informal nature of TikTok-style text. In contrast, TF-IDF combined with a Linear SVM delivered the most reliable and stable performance in terms of accuracy, precision, recall, F1-score, and even training time.

After selecting the optimal configuration, the final model was trained, calibrated, and enhanced with additional scoring mechanisms such as confidence and credibility scores to create a more informative output. The last part of the chapter outlined how the entire machine-learning pipeline was deployed through a lightweight Flask API. The API mirrors the training-time workflow by cleaning the input, vectorizing it, generating the prediction, extracting potential sources, and returning the results in a structured format. With this setup, the system responds to user input in almost real time, completing the full process in around five seconds. Altogether, Chapter 5 demonstrated how System 1 moved from concept to a fully integrated, fast, and practical component of the Faktz platform.

Chapter 6: Testing and Evaluation

6.1 Overview

Chapter 6 focuses on evaluating how the model performs in practice and whether it meets the goals that guided its development. After completing the full machine-learning pipeline and integrating it into the Faktz platform, the system was tested using a structured set of procedures designed to measure accuracy, consistency, stability and real-world usability. This chapter brings together the results of these tests and explains what they reveal about the model's behavior, its strengths and its limitations.

The evaluation process begins by assessing the quality of the final classifier through several metrics, including accuracy, precision, recall and F1-score. These provide a balanced view of how well the model distinguishes among the three credibility categories. Additional tests explore how the system behaves across different types of statements, ensuring that performance remains reliable for short, informal and highly varied user input. The chapter also examines the confidence-score calibration to confirm that both operate as intended and add meaningful value to the classification output.

Finally, the chapter evaluates the real-time performance of the deployed API, testing execution speed, stability under repeated requests and its ability to return complete results within approximately five seconds. Together, these evaluation components offer a comprehensive understanding of how the model performs once deployed, showing how effectively it supports the overall objectives of the Faktz platform.

6.2 Evaluation Method

The performance of the model was assessed using a set of evaluation metrics that provide a balanced view of its behavior. Accuracy measured the overall percentage of correctly classified statements, while precision, recall, and F1-score offered deeper insight into how reliably each of the three labels was identified. A confusion matrix was also generated to visualize the distribution of correct and incorrect predictions across labels and to highlight any patterns of misclassification. Together, these metrics demonstrated the effectiveness of the proposed method and confirmed its suitability for deployment within the Faktz platform.

Table 8: Training Results based on Different Methods

Method	Training Time(s)	Training Accuracy	Training F1-Score
Bow	17.06	0.971	0.957
TF-IDF	1.01	0.967	0.962
Word2Vec	68.84	0.838	0.838
FastText	89.02	0.757	0.743

Table 9: Train VS Val Results

Method	Train Time (s)	Train Accuracy	Val Accuracy
Bow			
Logistic Regression	5.76	0.968	0.954
Random Forest	115.00	0.977	0.965
Linear SVM	17.06	0.971	0.957
TF-IDF			
Logistic Regression	1.20	0.956	0.954
Random Forest	118.91	0.977	0.967
Linear SVM	1.01	0.967	0.962

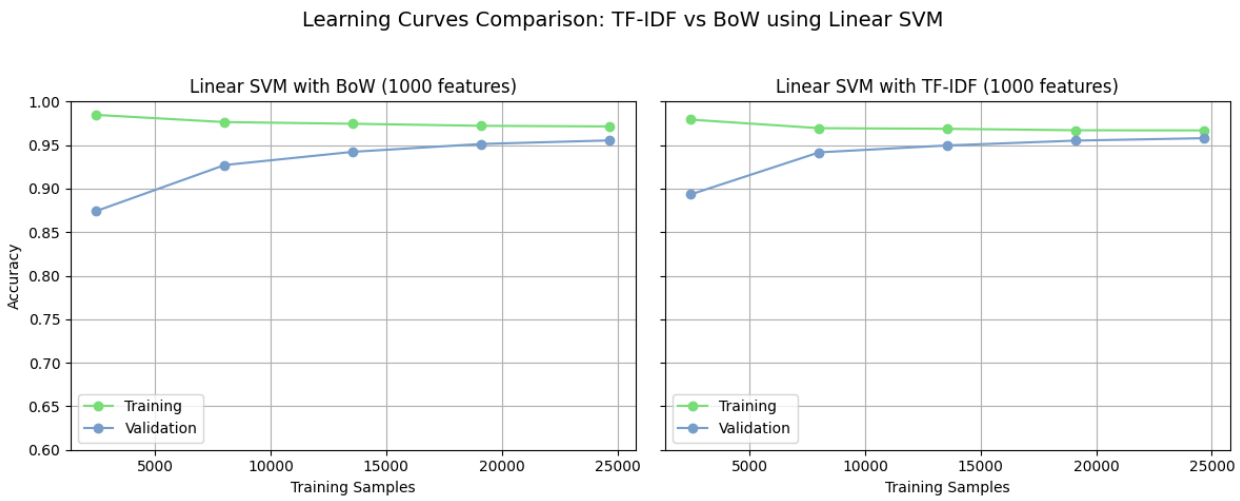


Figure 23: Train vs Val Learning Curve

6.3 Test Results

Table 10: Final Model Results

Dataset	Accuracy	F1-Score	Precision	Support	Recall
Combined	0.98				
Supported Claim	-	1	1	4535	1
Claim	-	0.98	0.98	5712	0.98
Opinion	-	0.97	0.97	2968	0.98

Confusion Matrix Interpretation

The confusion matrix provides a detailed view of how the model distinguishes between Supported Claim, Claim, and Opinion on the test set. The results show that the classifier performs consistently well across all classes, with only small amounts of misclassification. Supported Claims are recognized with very high accuracy, with 4,519 correctly identified and only 16 misclassified as Claims. No Supported Claim samples were incorrectly predicted as Opinions, demonstrating the model's ability to recognize statements that include a source or factual grounding.

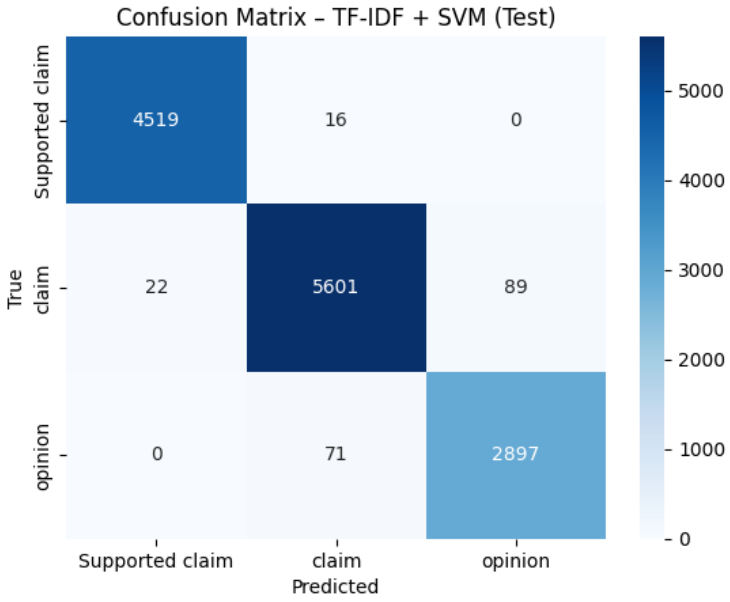


Figure 24: Confusion Matrix

The Claim category shows similarly strong performance, with 5,601 correct predictions. A small number of Claims were misclassified as Supported Claims (22) or Opinions (89). These errors are expected given the linguistic similarities between these two classes, especially when a claim is stated in a factual tone but does not explicitly reference a source. The Opinion class also displays high accuracy, with 2,897 correct predictions. Only 71 Opinion samples were incorrectly predicted as Claims, which may occur when subjective statements adopt assertive or factual phrasing.

Overall, the confusion matrix confirms that the TF-IDF + Linear SVM model handles all three labels effectively and maintains clear boundaries between them. The limited misclassification rates across categories demonstrate strong generalization and validate the suitability of the model for real-world use within the Faktz platform.

6.4 Real-Time System Testing

The final stage of evaluation involved testing the complete end-to-end pipeline to ensure that the AI model, Flask API, and web interface operated seamlessly together during live use. This testing process demonstrated how the system handles a real TikTok URL, extracts the transcript through the external API, processes the text through the classification pipeline, and returns the output to the user interface within a few seconds.

1. The Flask server was first launched in development mode, running locally on port 8000, making the /predict endpoint accessible for external testing tools such as Postman. A series of POST requests were then submitted to the endpoint using raw text as input. In these tests, the system successfully produced the expected structured JSON output, including the predicted label, confidence score, and extracted sources when detected. This confirmed that the deployed model responded correctly to external requests and maintained the same performance observed during offline evaluation.

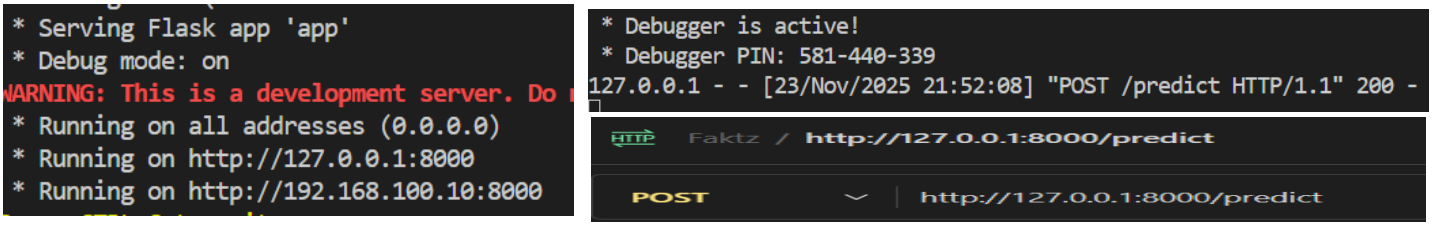


Figure 25: Testing the system in Flask Server

2. Then the system was tested using a full TikTok link directly through the Faktz website. The backend retrieved the transcript using the Supadata API, cleaned and preprocessed the text, applied the TF-IDF + SVM classifier, and returned a result with a high-confidence prediction. The server logs show the complete process from transcript retrieval to classification, executing in approximately 5.72 seconds as shown in figure 26 below. This confirms that the system meets the real-time requirement and can deliver fast, user-friendly analysis suitable for live deployment.

```
 * Debugger PIN: 977-571-886
Fetching transcript for: https://www.tiktok.com/@sideyou0/video/7505506425733483798?q=health%20facts&t=1763493298044
Supadata raw response: {'lang': 'en', 'availableLangs': ['en'], 'content': "This is what cervical cancer looks like inside the body. It starts slowly in the cervix, the lower part of the uterus that connects to thehe body and he body and starts to change the cells in the cervix. These damaged cel
he body and starts to change the cells in the cervix. These damaged cells can grow out of control, and that's how cancer begins. In this 3D view, you can see how the virus enters the cervix and how a small change becomes something dangerous. At first, there are no symptoms. No pain, no bleeding, nothing. That's why it's called a silent killer. Later, some women may notice bleeding after sex, pain in the lower belly, or strange discharge. But by then, it might already be serious. The good news is cervical cancer can be prevented. Regular checkups like pap smears and the HPV vaccine can save lives. Early detection means easy treatment. This cancer grows slowly, but if ignored, it can spread to other parts of the body. So don't wait. Your body gives signs. Listen to them. Share this video. It could save someone's life."}
Transcript and metadata fetched successfully.
Analysis completed in 5.72 seconds
127.0.0.1 - - [23/Nov/2025 20:57:53] "POST /predict HTTP/1.1" 200 -
```

Figure 26: Backend Run Time Proof

3. The final output displayed on the frontend matched the backends' response exactly, with consistent confidence scores above 98%, accurate label predictions, and proper rendering of the analysis dashboard. This real-time test demonstrates that the integration between the AI model, API, and web interface is stable, reliable, and ready for practical use within the Faktz platform.

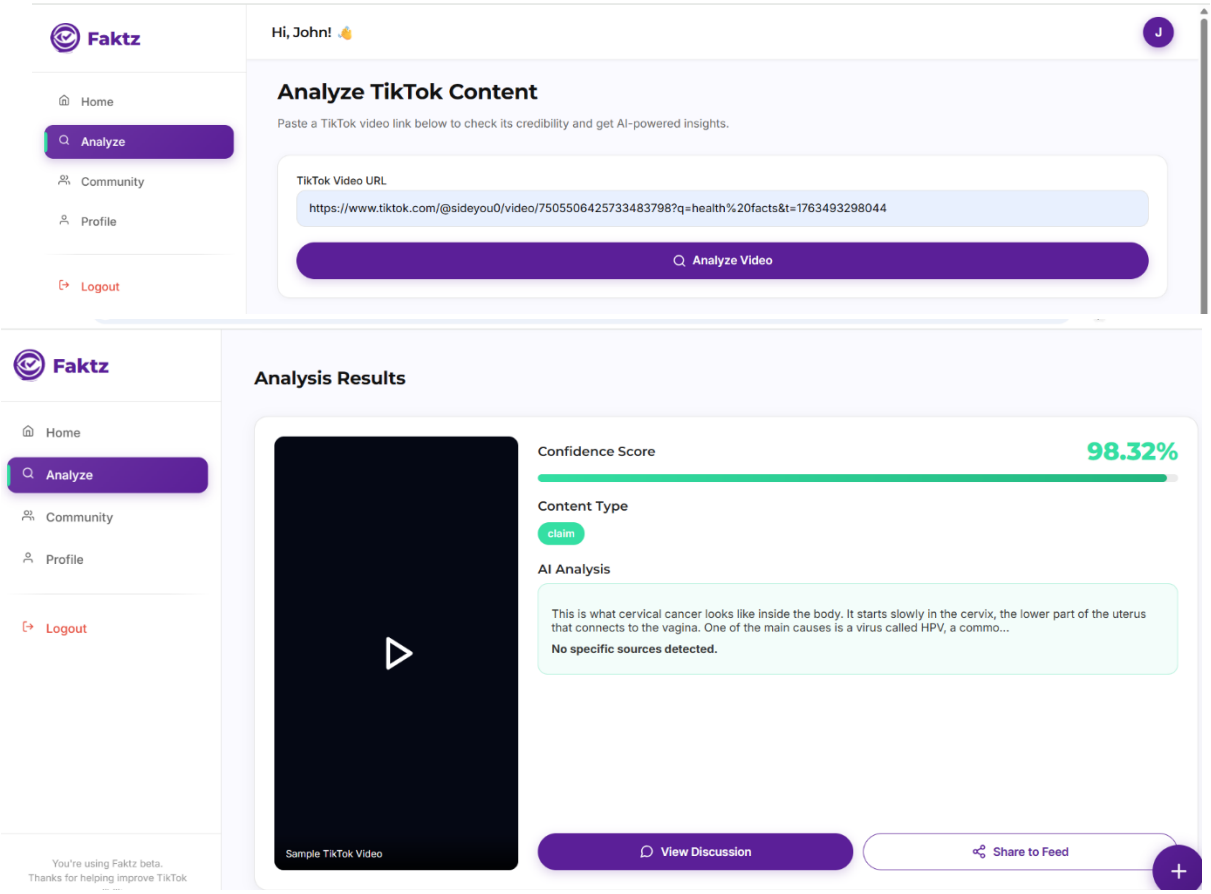


Figure 27: Website Integration

6.5 Summary

The testing and evaluation phase provided a comprehensive understanding of how the proposed credibility-classification system performs under both controlled and real-world conditions. Offline evaluation demonstrated that the TF-IDF and Linear SVM combination consistently outperformed other model configurations, delivering strong accuracy, balanced precision and recall, and stable F1-scores across all three classes. The confusion matrix confirmed that misclassifications were minimal and largely limited to linguistically ambiguous cases, indicating that the model developed a clear understanding of the distinctions between Supported Claims, Claims, and Opinions.

Beyond offline evaluation, the real-time tests played an essential role in validating the system's readiness for deployment. Testing through Postman verified that the Flask API correctly processed external requests and returned structured predictions. Full end-to-end testing through the Faktz web interface demonstrated the system's ability to handle a complete TikTok URL, retrieve the transcript through the Supadata API, run the text through the full preprocessing and classification pipeline, and return a final prediction that is immediately displayed to the user. The pipeline consistently achieved analysis times of around five seconds, confirming that the system meets its performance requirements for a real-time user experience.

Overall, the results from both offline experimentation and real-time testing provide strong evidence that the proposed model is accurate, efficient, and reliable. The system performs well across diverse input types, integrates smoothly with the web platform, and consistently produces high-confidence, interpretable outputs suitable for practical use in enhancing the credibility awareness of TikTok users.

Chapter 7: Conclusion

7.1 Overview

This chapter brings the project to its final point by summarizing the full journey from the initial problem definition to the completed and tested system. Throughout the earlier chapters, the work gradually moved from understanding the credibility challenges on TikTok, to designing a suitable solution, to building and evaluating a complete AI-driven classification system. The final outcome is a fully functioning pipeline that can analyze TikTok content, classify it as Supported Claim, Claim, or Opinion, and return a confidence score and interpreted result directly to the user.

The system now operates smoothly within the Faktz platform, allowing users to enter a TikTok URL and receive an analysis in around five seconds. This includes transcript retrieval, preprocessing, classification, and source extraction when applicable. The strong performance of the TF-IDF and Linear SVM model, supported by extensive dataset preparation and multiple rounds of testing, demonstrates that the system is both accurate and practical for real-world use. The next sections present the user manual, discuss the significance of the system, highlight its constraints, and outline directions for future enhancement.

7.2 User Manual

This section provides a simple guide for users to interact with the credibility-classification system integrated into the Faktz platform. The aim is to ensure that anyone, regardless of technical background, can understand how the system works and how to obtain accurate credibility insights from TikTok content.

1. Accessing the Faktz Platform

Users begin by navigating to the Faktz web platform through a standard web browser. The main dashboard provides access to the analysis page, community features, and user profile.

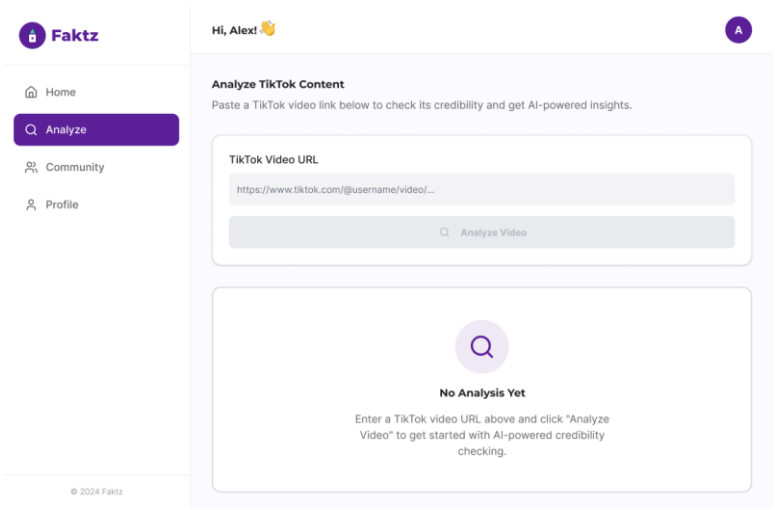
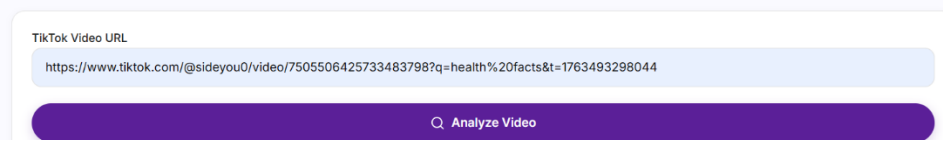


Figure 28: Faktz Analyze Page

2. Submitting a TikTok Video

To analyze a video, the user pastes a complete TikTok URL into the input field. The link can point to any public TikTok video, including educational, news-related, or general informational content.



3. Running the Analysis

After inserting the link, the user presses the Analyze Video button. This triggers the backend pipeline, which:

- Retrieves the transcript using the Supadata API
- Preprocesses the extracted text
- Passes the cleaned content through the TF-IDF + Linear SVM model
- Computes the confidence score
- Performs source extraction if references are detected

The entire analysis process is normally completed in around five seconds.

4. Viewing the Results

Once the processing is finished, the results are displayed directly on the same page.The output includes:

- **Confidence Score:** A percentage that shows how certain the model is about prediction
- **Content Type:** The assigned label (Opinion, Claim, or Supported Claim)
- **AI Analysis:** A breakdown including detected source references (if present)
- **Video Preview:** A thumbnail of the analyzed TikTok content

5. Interpreting the Output

- If the label is **Opinion**, the statement reflects personal views rather than factual claims.
- If the label is **Claim**, the text presents a fact-like assertion but without referenced evidence.
- If the label is **Supported Claim**, the statement includes a verifiable source or reference.

Users can rely on the confidence score to understand how certain the model is about its categorization.

6. Recommended Use

The system is intended as a credibility-checking assistant rather than a replacement for professional fact-checking. It helps users make more informed judgments about the information they encounter on TikTok by offering quick, automated insights.

7.3 Significance

The significance of this project goes beyond building a working machine-learning model; it lies in the role the Faktz platform can play in helping people pause and think more carefully about the information they encounter online. TikTok has become one of the fastest-growing spaces for news, explanations, opinions and claims, yet users often scroll through content so quickly that they rarely question its accuracy. The system developed in this project offers a simple and accessible way to slow that process down. By classifying content into Opinion, Claim or Supported Claim, and by presenting clear confidence score the platform encourages users to take a moment to reflect on what they are watching rather than accepting it at face value.

The impact of this becomes even stronger when combined with the community aspect of Faktz. The platform is not only a tool that returns a label; it gives people a space to share what they discover, discuss questionable content, and learn from each other's analysis. This transforms the experience from a private check into a shared practice of thinking critically and holding information to a higher standard. As a result, the system supports healthier digital habits by helping users build the habit of "thinking twice" before trusting or spreading information.

On a technical level, the project demonstrates how carefully prepared data, well-chosen modelling methods and thoughtful system integration can come together to address real challenges in modern information environments. On a social level, it contributes to digital literacy by giving users an intuitive, supportive tool that encourages responsible media consumption. In this way, the significance of Faktz lies not only in its technical success but in its potential to influence how people understand and interact with the content they consume every day.

7.4 Constraints

Despite the strong performance and successful deployment of the system, several constraints were identified throughout the development and testing stages. These constraints do not diminish the value of the work but provide important context for understanding its current limitations and the opportunities they create for future improvement.

One key constraint relates to the nature of the dataset itself. Although the dataset was significantly expanded using synthetic and restructured news samples, the system still relies heavily on text-only inputs. TikTok videos often contain visual cues, background audio, or non-verbal signals that influence the meaning of a statement, and these elements are not yet captured by the current model. As a result, the system focuses solely on the transcript and may overlook credibility cues that exist outside the text.

Another constraint arises from the informal and fast-changing style of language used on TikTok. Slang terms, newly emerging expressions, and creative sentence structures may not always be captured within the TF-IDF vocabulary, leading the model to occasionally miss subtle linguistic signals. Although the system performs well overall, the constant evolution of social-media language means that periodic retraining or vocabulary updates may be necessary to maintain long-term accuracy.

Finally, the platform depends on external APIs, such as Supadata to retrieve TikTok transcripts. This introduces limitations related to rate limits, availability, or temporary failures on the external provider's side. While the system includes checks to ensure stable operation, the dependency remains an unavoidable constraint for real-time transcript retrieval.

Overall, these constraints do not affect the core contribution of the project but serve as important considerations for interpreting results and planning future enhancements.

Table 11: Constraints

Constraints
Relies on text-only transcripts, without analyzing visuals or audio.
TikTok language evolves quickly, which may require periodic retraining.
Vocabulary coverage limitations for slang or newly emerging phrases.
Depends on external APIs for transcript retrieval, which may introduce reliability issues.
Supported Claim detection identifies source names but does not verify their credibility.

7.5 Future Enhancements

Looking ahead, there are several meaningful ways the Faktz system can grow and become even more helpful for users. One of the most important improvements involves expanding the model beyond text-only analysis. TikTok often carries meaning through visuals, tone of voice, editing style and background audio, and adding these elements into the analysis would give a much richer picture of how credible a video actually is. A multimodal version of the system could understand not only what the creator says but how they say it, which would make the credibility assessment more accurate and more aligned with how users naturally interpret content.

Another valuable direction is improving the way sources are handled. At the moment, the system can detect when a statement mentions a source, but future versions could go further by checking whether that source is legitimate, trustworthy or even relevant to the claim being made. This would turn the platform into a much stronger credibility assistant, helping users recognize not only when a creator cites a source but whether that source actually supports the statement or is being misused.

The system would also benefit from a dynamic, continuously updated dataset. TikTok language changes quickly, with new slang, trends and formats appearing all the time. Allowing the model to retrain periodically or even learn from examples shared by the Faktz community would keep it aligned with the platform's fast-paced nature. This would make the tool feel more responsive and accurate as online language evolves.

Improving the explainability of predictions is another meaningful enhancement. Highlighting the key phrases that influenced the model's classification or giving a short, understandable rationale could help users learn why a statement was flagged as an Opinion, Claim or Supported Claim. This encourages critical thinking and helps users develop healthier habits around consuming online information.

Finally, strengthening the system's backend infrastructure will allow Faktz to scale smoothly as more people use it. Faster servers, better load balancing and more efficient deployment pipelines will ensure that users continue to receive results in around five seconds, even under heavy usage.

Together, these enhancements would make Faktz not just a classification tool, but a genuinely supportive space that helps people pause, reflect and engage more thoughtfully with the content they encounter every day.

Table 12: Future Enhancements

Future Enhancements
Integrate multimodal analysis (video frames, audio cues, tone detection).
Implement source verification to check if a cited source is legitimate or relevant.
Build a dynamic, continuously updated dataset that adapts to new slang and trends.
Add explainability features (highlight key phrases influencing classification).
Improve backend scalability for faster and more robust real-time performance.

7.6 Summary

This project set out to design and develop an AI-powered system capable of analyzing TikTok content and helping users better understand the credibility of the information they encounter online. Across the chapters, the work progressed from problem identification to dataset construction, model development, system integration and real-time evaluation. The final system successfully classifies TikTok text into three categories using a TF-IDF and Linear SVM model that demonstrated strong accuracy, fast inference speed and consistent performance across evaluation metrics.

Beyond technical achievement, the project explored the social importance of promoting digital awareness and encouraging users to pause and reflect on the content they consume. The Faktz platform brings these elements together by offering not only an automatic credibility-checking tool but also a space where users can share results, discuss findings and learn from one another. Although several constraints were identified, such as reliance on text-only inputs and the limitations of external transcripts, these did not hinder the system's core objective. Instead, they open meaningful opportunities for future work, including multimodal analysis, source verification and adaptive dataset updates.

Overall, the project achieved its intended goals and produced a functioning solution that is both practical and impactful. The system stands as a solid foundation for further expansion and has the potential to continue evolving alongside the fast-changing nature of online content.

Table 13: Faktz: An NLP and Machine Learning-Based Platform for Content Credibility on TikTok

Project Objective	Research Question(s)	Outcome	Remark / Future Enhancements
To identify and define appropriate credibility labels that effectively differentiate TikTok content based on evidence, linguistic cues, and communication style.	Can conceptual and linguistic criteria distinguish speech in short-form videos?	Three clearly defined labels were established (*Opinion*, *Claim*, and *Supported Claim)* based on evidence patterns, language structure, and source presence. These labels guided dataset preparation and model design.	Expand label definitions to include subcategories (e.g., misleading claim, satire) for richer credibility interpretation.

To collect, clean, and annotate TikTok transcripts to build a reliable, high-quality dataset for credibility classification.	Which data sources and collection methods best capture the language style of TikTok?	A hybrid dataset combining Kaggle TikTok data, synthetic template-generated samples, and restructured news content was created. Cleaning preserved TikTok-style informality while ensuring consistency. Annotation rules successfully separated the three classes.	Introduce automated dataset updates, community-driven submissions, and continuous retraining to adapt to evolving TikTok slang and new misinformation trends.
	What cleaning and preprocessing techniques ensure consistency while preserving informal structure?		
	How can annotation guidelines clearly separate the labels?		
To develop and train an NLP-based ML model to classify TikTok transcripts into credibility categories.	Which NLP and ML models best fit this task?	TF-IDF combined with Linear SVM outperformed alternatives (BoW, Word2Vec, FastText, Random Forest, Logistic Regression). The model achieved high accuracy, balanced precision/recall and stable convergence.	Explore multimodal models that incorporate visual cues, tone, audio, and metadata to improve real-world classification accuracy.
	What linguistic features strengthen separation between the three classes?		
To evaluate model performance and system usability through validation metrics and user feedback.	Which evaluation metrics best reflect real-world performance?	Evaluation using accuracy, precision, recall, F1-score and confusion matrices confirmed strong performance. Sample predictions and real-time testing validated system usability.	Add explainability features (highlight key words, provide classification rationale) and integrate lightweight user-feedback loops.

To deploy the trained model into a web-based platform.	How can credibility classifications be displayed in a user-friendly format?	The model was deployed via a Flask API and fully integrated into the Faktz web app, delivering complete analysis — including label, confidence score and sources — in ≈5 seconds.	Improve backend scalability and add source-credibility checking to evaluate whether referenced sources are reliable or relevant.

References

1. Kirkpatrick, C. E., & Lawrie, L. L. (2024). TikTok as a Source of Health Information and Misinformation for Young Women in the United States: Survey Study. JMIR infodemiology, 4, e54663. https://doi.org/10.2196/54663
2. Navlakha, M. (2023). Which countries have banned TikTok? [online] Mashable SEA. Available at: https://sea.mashable.com/tech/22984/which-countries-have-banned-tiktok.
3. Bushak, L. (2022). Nearly 84% of mental health videos on TikTok are misleading: study. [online] MM+M - Medical Marketing and Media. Available at: https://www.mmm-online.com/home/channel/nearly-84-of-mental-health-videoson-tiktok-are-misleading-study/
4. Schultz, K. (2025). Understanding the Role of Knowledge Intelligence in the CRISP-DM Framework: A Guide for Data Science Projects - Enterprise Knowledge. [online] Enterprise Knowledge. Available at: https://enterprise-knowledge.com/understanding-the-role-of-knowledge-intelligence-in-the-crisp-dm-framework-a-guide-for-data-science-projects/
5. Pereira, B.B., Ha, S., Ha, S. and Ha, S. (2024). ENVIRONMENTAL ISSUES ON TIKTOK: TOPICS AND CLAIMS OF MISLEADING INFORMATION. *Journal of Baltic Science Education*, [online] 23(1), p.Continuous. doi: https://doi.org/10.33225/jbse/24.23.131 .
6. Cools, K., Gideon and Maathuis, C. (2024). *Modeling offensive content detection for TikTok*. [online] doi: https://doi.org/10.48550/arXiv.2408.16857 .
7. Shang, L., Zhang, Y., Deng, Y. and Wang, D. (2025). MultiTec: A Data-Driven Multimodal Short Video Detection Framework for Healthcare Misinformation on TikTok. IEEE Transactions on Big Data, pp.1–18. doi: https://doi.org/10.1109/tbdata.2025.3533919 .
8. Sokolova, K. and Kefi, H. (2020). Instagram and YouTube Bloggers Promote it, Why Should I buy? How Credibility and Parasocial Interaction Influence Purchase Intentions. Journal of Retailing and Consumer Services, 53(1). doi: https://doi.org/10.1016/j.jretconser.2019.01.011 .

9. Capitol Technology University (2023). TikTok and the War on Misinformation | Capitol Technology University. [online] www.captechu.edu. Available at: https://www.captechu.edu/blog/tiktok-and-war-misinformation.
10. Belanche, D., Casaló, L.V., Flavián, M. and Sánchez, S.I. (2021). Understanding Influencer marketing: the Role of Congruence between influencers, Products and Consumers. Journal of Business Research, [online] 132(1), pp.186–195. doi: https://doi.org/10.1016/j.jbusres.2021.03.067 .
11. Tiktok.com. (2024). Safety partners. [online] Available at: https://www.tiktok.com/safety/en/safety-partners .
12. Tiktok (2024). Integrity and Authenticity. [online] Tiktok.com. Available at: https://www.tiktok.com/community-guidelines/en/integrity-authenticity.
13. Questionnaire for Fact-Finding – 'Faktz' AI Credibility Platform. [online] Google Docs. Available at: https://docs.google.com/forms/d/e/1FAIpQLScHmkDahdbijVdb4sRmnjZYmhecBmc0FfP4RKWwO-L13pnhiQ/viewform?usp=header

Appendices

Appendix A: Questionnaire Structure and Questions

Section A: TikTok Usage Patterns

No.	Question	Purpose	Options	Question Type
1	How often do you use TikTok?	To determine the frequency of platform usage and establish baseline user behaviour.	Daily, Several times a week, Weekly, Rarely	Multiple-choice
2	How much time do you spend scrolling on TikTok per day?	To measure daily exposure to TikTok content.	Less than 30 minutes, 30–60 minutes, 1–2 hours, More than 2 hours	Multiple-choice
3	What type of content do you engage with the most?	To identify content categories most associated with claims, opinions, and misinformation.	Health, Fitness, Politics, Entertainment, Lifestyle, Celebrity News, Other	Multiple-choice

Section B: Exposure to Claims and Misinformation

No.	Question	Purpose	Options	Question Type
4	How often do you see videos that present “facts,” “advice,” or “tips”?	To assess exposure to content that may include unsupported claims.	Very Often, Often, Sometimes, Rarely, Never	Multiple-choice
5	Do you usually check whether such information is true before believing or sharing it?	To understand verification habits and the likelihood of misinformation spread.	Always, Sometimes, Rarely, Never	Multiple-choice
6	Have you ever believed a video that turned out to be misleading?	To measure direct experience with misinformation.	Yes, No, Not sure	Multiple-choice

Section C: Perception of Misinformation and Influencer Responsibility

No.	Question	Purpose	Options	Question Type
7	Which topic do you think spreads the most misinformation?	To identify the content areas most associated with misleading information.	Health, Fitness, Politics, Education, Lifestyle, Sports, Celebrity News, Other	Multiple-choice
8	Why do you think some influencers spread misinformation?	To understand perceived motivations behind spreading inaccurate information.	For more engagement, Lack of knowledge, Intentional clickbait, No strict platform rules, Paid sponsorships, Other	Multiple-choice
9	What do you think would help influencers share more accurate content?	To identify user perspectives on solutions for reducing misinformation.	Credibility checks, Penalties, Rewards for accurate content, Collaboration with experts, Education	Multiple-choice

Section D: Perception of AI-Based Credibility Tools

No.	Question	Purpose	Options	Question Type
10	How confident are you in AI's ability to detect misinformation accurately?	To measure trust in AI systems.	Scale of 1 to 5	Likert-scale
11	Would you use a platform like Faktz that labels videos as Opinion, Claim, or Supported Claim?	To determine willingness to adopt the system.	Definitely, Most probably, Not sure, No	Multiple-choice
12	How helpful would an AI confidence score be to you?	To evaluate the value of explainability features.	Scale of 1 to 5	Likert-scale

Section E: Community Engagement and Expert Involvement

No.	Question	Purpose	Options	Question Type
13	Would you trust the platform more if verified professionals were active?	To assess the importance of expert contributions.	Definitely, Most probably, Not sure, No	Multiple-choice
14	How likely are you to participate in discussions about video credibility?	To evaluate potential engagement levels.	Scale of 1 to 5	Likert-scale
15	What feature would make you feel safest contributing to discussions?	To identify user expectations regarding platform safety.	Clear rules, Moderation, Verified experts, Anonymous posting, Other	Multiple-choice

Appendix B: Dataset Sample

	claim_status	video_transcription_text
0	claim	elect Donald Trump this week, but only after o...
1	Supported claim	Reuters published an article saying BERLIN The...
2	opinion	in our opinion humans physically cannot hum wh...
3	Supported claim	Reuters published an article saying KAMPALA Ug...
4	opinion	my colleagues think that it is illegal to wast...

Appendix C: Pre-Viva Poster

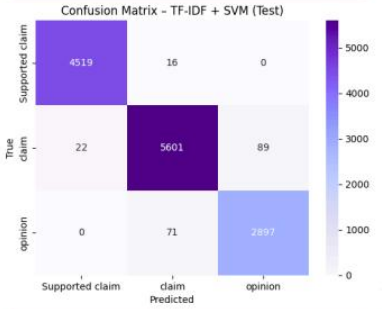
Faculty of Data Science and Information Technology (FDSIT)

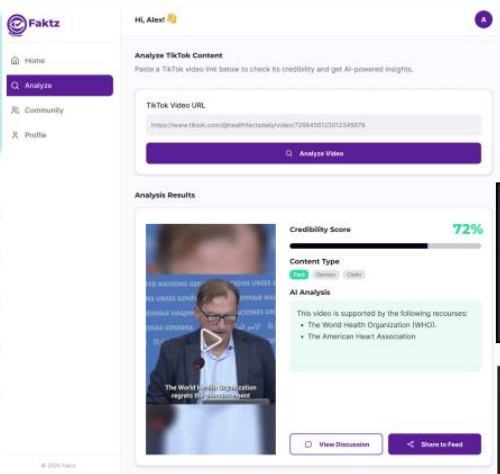
INTI International University

Faktz: An NLP and Machine Learning-Based Platform for Content Credibility on TikTok

Problem Statement

- TikTok's algorithm prioritizes engagement, not accuracy.
- Misinformation on TikTok spreads faster than users can verify it.
- There is no tool that identifies unverified content.





System Strengths

- Trained on 45k+ TikTok + news transcripts
- Handles Gen Z slang & informal language
- Fast pipeline (clean → vectorize → classify)
- High-accuracy SVM classifier
- Transparent confidence scoring

Test Scores

```
Test Accuracy: 0.985017026106697
              precision    recall  f1-score   support

Supported claim       1.00      1.00      1.00      4535
       claim          0.98      0.98      0.98      5712
     opinion          0.97      0.98      0.97      2968

    accuracy                             0.99     13215
   macro avg          0.98      0.98      0.98     13215
weighted avg          0.99      0.99      0.99     13215
```

Input

```
"text": "According to a 2023 report from the World
    Health Organization (WHO), regular physical
    activity can reduce the risk of heart disease by
    up to 30%. The American Heart Association also
    recommends at least 150 minutes of moderate
    exercise per week, confirming that staying active
    has long-term health benefits for the heart and
    overall wellbeing."
```

Output

```
{
  "confidence (%)": 98.89,
  "label": "Supported claim",
  "resource": "Resources: the World Health Organization
    (WHO, The American Heart Association"
```

Project Objectives

- Collect, clean, and annotate TikTok transcripts
- Build a complete NLP processing pipeline.
- Train an ML classification model for credibility.
- Deploy the model into a web platform.

System Features

01 TikTok Transcript Processing Pipeline

02 ML-Powered Credibility Classification

03 Source Extraction for Supported Claims

04 Transparent Confidence Score Output

Student's Deatils

Mayar Abu Latifa
ID: I25036233

Program: Bachelor's of Information Technology
Email: i25036233@student.newinti.edu.my

Acknowledgements

Special thanks to my supervisor, Dr. Atif Mahmood, for his continuous support, guidance, and prompt assistance throughout this project.