PsychExtract: Preliminary Report

COURSE
CM3070: Final Project
Computer Science
University of London

6011 words

ABSTRACT

Psychotherapeutic practice increasingly relies on reflective analysis of client narratives, yet manual extraction of emotional and cognitive patterns from written material remains time-intensive and subjective. PsychExtract investigates the feasibility of an AI-assisted pipeline designed to support psychotherapeutic reflection by automatically extracting emotional signals, insight-related themes, and interpretable summaries from client-generated text. The system integrates optical character recognition (OCR) for handwritten and scanned journal entries, natural language processing (NLP) techniques for fine-grained emotion classification and thematic extraction, and an output layer that provides both textual and text-to-speech (TTS) audio summaries. This preliminary report grounds the system design in established psychological theory, drawing on research into insight as a mechanism of therapeutic change and the role of emotion in cognitive restructuring. Existing AI-supported mental-health tools are reviewed to identify methodological and ethical constraints, informing the system's design principles and evaluation strategy. The implemented architecture combines OCR preprocessing, transformer-based emotion classification inspired by the GoEmotions framework, keyword and topic extraction methods for interpretability, and text-to-speech synthesis to improve accessibility. A functional prototype is developed with particular emphasis on OCR feasibility, as transcription quality directly affects all downstream analyses. The prototype is evaluated using quantitative metrics and qualitative error analysis to assess recognition performance, preprocessing effects, and system robustness. Results demonstrate that while OCR performance varies across handwriting styles, targeted preprocessing and error analysis substantially improve downstream interpretability. The findings highlight both the promise and limitations of deploying NLP-driven analysis in sensitive mental-health contexts, and motivate future work on multimodal evaluation, user-centred validation, and clinical collaboration.

CONTENTS

INTRODUCTION

Psychotherapy relies heavily on the ability to recognise emotional cues, behavioural patterns, and subtle linguistic markers that reveal how a client is progressing. Yet the primary documents containing these signals (handwritten notes, personal reflections, and informal therapeutic writings) remain largely unstructured, inconsistently interpreted, and rarely analysed beyond the immediate therapeutic context. Manual coding of these underlying insights is labour-intensive and subjective, leaving a significant amount of potentially meaningful information unused. As digital mental health tools continue to expand, there is an opportunity to explore whether computational techniques can support practitioners by providing structured, transparent summaries of this material without replacing clinical judgement.

PsychExtract is developed in response to this need. The system investigates how handwritten mental-health-related text can be transformed into structured psychological insight through a modular pipeline combining optical character recognition (OCR), principled Natural Language Processing (NLP) preprocessing, interpretable transformer-based emotion classification, lightweight linguistic analysis, and optional text-to-speech (TTS) for accessibility. While the system does not aim to conduct therapy or interpret clients, it seeks to support mental health professionals, psychology students, and researchers by offering clearer, standardised representations of unstructured text that may assist reflection, case formulation, or comparison across documents.

A key methodological decision in this project is the choice to adopt Template 4.1 (orchestrating AI models to achieve a goal). Because several components of the pipeline (OCR, emotion classification, interpretability layers, and TTS) offer many viable model options, the project naturally lends itself to a comparative approach. Template 4.1 provides an explicit structure for integrating and evaluating interchangeable models within a coherent system. This allows the project to determine which combinations produce the most accurate, transparent, and usable workflow, while remaining grounded in the needs of psychological practitioners and researchers.

To understand what kinds of insight such a system might surface, the project draws from established work on the role of emotion and insight in psychotherapy [1, 2], existing AI-supported therapeutic interfaces [3], NLP methods for analysing unstructured text [5, 6], and input–output tools relevant to this domain [8, 9]. These foundations, reviewed in the Literature Review, highlight both the clinical relevance of emotional and linguistic cues and the current lack of tools that can extract them from handwritten sources in a transparent and practitioner-friendly way.

The early development of PsychExtract takes a user-centred perspective suitable for human–computer interaction research. The aim is not to produce a deployable system but to explore which design features, model configurations, and forms of output best align with the needs of the intended audience. The overall pipeline and its justification are presented in the Design section of this report, which outlines the domain, intended users, system features, and feasibility considerations. Because OCR accuracy has a substantial downstream effect on the entire pipeline, the first stage of prototyping focuses exclusively on OCR. This early focus provides a stable foundation for subsequent stages of development.

The Prototyping component of this report describes this initial OCR prototyping stage, including curated handwritten samples, preprocessing procedures, and early observations. While full evaluation and downstream component testing fall outside the scope of this preliminary report,

the prototyping section outlines the next steps for expanding the pipeline and continuing model comparison.

LITERATURE REVIEW

FOUNDATIONS OF INSIGHT AND EMOTION IN PSYCHOTHERAPY

Insight as a Therapeutic Mechanism

Insight is widely recognised as a core driver of psychological change. Hill et al. describe it as a "conscious meaning shift involving new connections" [1, p. 442], noting that early insights often begin as simple realisations before deepening into more complex higher-dimensional forms (such as emotional understanding, cognitive restructuring, and reframing past experiences). They emphasise that the concept lacks a universally accepted definition and varies across therapeutic approaches. This ambiguity positions insight as both central and flexible, making it suitable for computational modelling when carefully scoped.

For the purposes of this project, these theoretical limitations (lack of consensus, variation across schools, and multi-dimensionality) clarify which components can be practically extracted from text. PsychExtract therefore focuses exclusively on extracting early-stage insight, which is typically expressed through observable language patterns such as emotional descriptions, reflective statements, and self-evaluative comments. These early meaning shifts are the aspects of insight most consistently expressed through text and most amenable to natural language analysis.

This contextualises why insight extraction matters. If early insight influences therapeutic progress, then summarising indicators of such insight from client reflections can help therapists track meaningful shifts over time. This first section of literature review establishes the theoretical motivation for the system, that is, insight is important, language is one of the primary ways it appears, and early insight is computationally detectable.

Emotion as a Core Component of Therapeutic Change

Greenberg and Pascual-Leone argue that emotional processing is a primary driver of change across therapeutic modalities [2]. They outline a process involving emotional awareness, regulation, transformation, and meaning-making. This typically requires clients to articulate internal emotional experiences. While therapists are trained to detect emotional cues, much emotional content is communicated implicitly through language, making consistency and standardization difficult in practice.

This challenge motivates PsychExtract. If written reflections, such as journals, homework tasks, and progress updates, contain emotional signals that contribute to therapeutic insight, then automated extraction of emotional and linguistic patterns can support therapists by ensuring these signals are made visible and consistently interpreted. The aim is not to replace therapist judgement. Rather, PsychExtract produces structured summaries of emotional and cognitive patterns derived from client text. These summaries can draw attention to potential therapeutic themes without making clinical claims, maintaining alignment with ethical guidance in the field. This naturally leads to the next question of whether artificial intelligence is a suitable tool for supporting therapists in recognising these linguistic signals.

AI Support in Psychotherapy: Lessons from Existing Systems

The use of artificial intelligence in mental-health contexts is not new. DeVault et al. introduced SimSensei [3], a virtual interviewer designed to detect psychological distress from verbal and nonverbal behaviour. Their work demonstrates several key findings relevant to PsychExtract. Notably, people often disclose more openly when interacting with automated systems, and even simple computational methods can highlight meaningful psychological cues (such as sentiment shifts or linguistic markers of distress).

Figure 1. SimSensei virtual inteviewer

SimSensei's limitations are equally informative. Because it operates in real-time conversation, it must use extremely cautious and overly simplistic natural language models to avoid unsafe or inappropriate responses. As a result, the system relies on basic language processing.

PsychExtract diverges from this setting in two important ways. First, it is non-conversational. Users provide reflective text, and the system produces an analysis, not an ongoing dialogue. Second, it does not operate in real-time. These affordances allow PsychExtract to employ more advanced language-processing techniques safely, such as transformer-based architectures like BERT, because there is no risk of generating incorrect or harmful conversational replies.

This section therefore establishes why artificial intelligence is appropriate for insight extraction. Existing work shows that AI can highlight clinically relevant linguistic cues, and PsychExtract extends this by applying stronger models in a safer, offline workflow. This bridges into the next section by motivating how AI can be used. This is through specific natural language processing methods tailored to the linguistic components that make up early insight.

NLP Methods for Emotion, Insight, and Cognitive Pattern Extraction

Before examining technical methods, it is important to clarify terminology for non-specialist readers. NLP refers to computational techniques for analysing or generating human language. Modern NLP often uses transformer-based models, which are deep learning architectures capable of understanding words in context rather than in isolation. These models outperform traditional techniques in tasks involving emotion recognition, topic inference, and meaning extraction, all of which are relevant to early insight.

This section details the three components of insight that PsychExtract identifies through NLP: Emotion expression, cognitive themes and reflective topics, and linguistic patterns associated with meaning-making.

By connecting these components to the earlier theory section, PsychExtract grounds its extraction pipeline directly in the psychological mechanisms of insight.

Fine-Grained Emotion Classification (GoEmotions)

Demszky et al. introduce GoEmotions, a dataset of 58,000 Reddit comments labelled with 27 fine-grained emotion categories excluding a neutral class [4]. Their findings show that transformer-based models such as BERT significantly outperform traditional machine learning approaches for understanding emotional nuance, especially because emotions often overlap and require contextual interpretation.

Positive		Negative		Ambiguous
admiration 👋	joy 😃	anger 😡	grief 😢	confusion 😕
amusement 😂	love ❤️	annoyance 😒	nervousness 😬	curiosity 🤔
approval 👍	optimism 🤞	disappointment	remorse 😔	realization 💡
caring 🤗	pride 😌	disapproval 👎	sadness 😞	surprise 😲
desire 😍	relief 😅	disgust 🤮		
excitement 🤩		embarrassment 😳		
gratitude 🙏		fear 😨		

Figure 2. GoEmotions affective labels

GoEmotions is valuable as a baseline for PsychExtract, but it has limitations. It contains short social-media comments rather than long reflective writing, and deeper therapeutic emotions (such as, grief processing, self-evaluation, growth-related fear) are underrepresented. To address this, PsychExtract uses GoEmotions models for initial benchmarking but extends beyond the dataset by incorporating long-form reflective text. This is in the form of available corpora (such as r/offmychest) or carefully synthesised paragraphs. This supports the system's goal of aligning emotion extraction with therapeutic contexts.

By grounding the emotional component of insight in this literature, PsychExtract builds directly on empirical evidence that transformer-based models are the strongest choice for contextual emotion detection. This sets the foundation for the next analytic component of understanding cognitive themes.

Cognitive Theme Extraction and Topic Representations (KeyBERT and BERTopic)

Cognitive themes represent the content of what clients reflect on. This is the issues, topics, meanings, and internal processes they describe. To extract these elements, PsychExtract evaluates two widely used NLP tools.

KeyBERT identifies keywords using semantic similarity between the text and candidate n-grams (varied word length groupings) [5]. Because it relies on Sentence-BERT embeddings, it captures meaning beyond simple word counts and is transparent enough to be interpretable by therapists. This makes KeyBERT a suitable, explainable baseline for cognitive theme extraction.

However, KeyBERT provides surface-level patterns and cannot capture broader shifts in meaning across a document. To complement this, PsychExtract includes a comparison with BERTopic, which identifies themes using clustering and class-based term frequency [6]. While more complex, BERTopic can represent broader reflective patterns that align with cognitive restructuring processes described in psychotherapy literature.

This therefore connects the "what" of insight (cognitive content) with the "how" of extraction (topic modelling methods), completing the second component of insight analysis. The next step is to capture linguistic patterns associated with meaning-making.

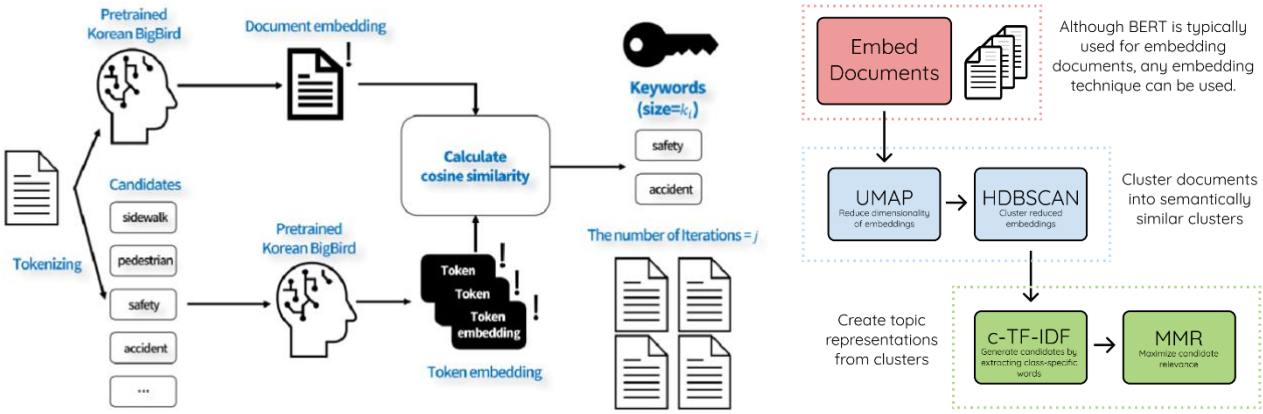

Figure 3: KeyBERT and BERTopic inner workings

Linguistic Pattern Analysis (LIWC)

The Linguistic Inquiry and Word Count (LIWC) framework categorises words into psychological dimensions such as cognitive processes, emotional tone, pronoun use, and insight-related terms [7]. Decades of studies demonstrate that these categories reflect internal cognitive states and are particularly relevant for detecting reflective thinking.

PsychExtract draws specifically on the cognitive mechanisms category. Words like "think," "realise," or "because" often signal reflective insight processes. However, LIWC is limited by its dictionary-based approach. It counts words without understanding context. This means it cannot distinguish between "I think" used casually versus reflectively.

To address this limitation, PsychExtract uses LIWC only for interpretability and theoretical grounding, while relying on contextual models (such as transformer-based NLP architectures) for the actual extraction pipeline. This hybrid approach supports interpretability without sacrificing nuance.

Having established how the system extracts early insight from text (emotion, cognitive themes, and linguistic patterns), the final section explains what text is fed into the system and how the results are returned to the user. This completes the OCR-NLP-TTS pipeline.

INPUT AND OUTPUT PROCESSING

OCR Requirements in Mental-Health Tools

In therapeutic settings, clients often maintain handwritten journals or written reflections. To analyse such inputs computationally, they must first be digitised using OCR, a technology that converts images of text into machine-readable characters.

Figure 4. OCR process

Smith provides a foundational overview of Tesseract, one of the most widely used open-source OCR engines [8]. Tesseract uses a multi-stage pipeline involving line detection, character segmentation, and language modelling to recognise text, even from noisy or imperfect inputs. However, Smith identifies two key limitations. For one, handwriting varies significantly between users, and for two, errors introduced by OCR can propagate into downstream NLP tasks, affecting emotion classification or topic modelling.

PsychExtract incorporates these findings by explicitly evaluating how OCR performance impacts the accuracy of insight-related NLP outputs. This extends prior OCR literature by shifting the focus from character-level accuracy to its influence on psychological inference quality. This is a critical factor in real-world mental-health tooling.

Output Processing: Text-to-Speech Synthesis of NLP Summaries

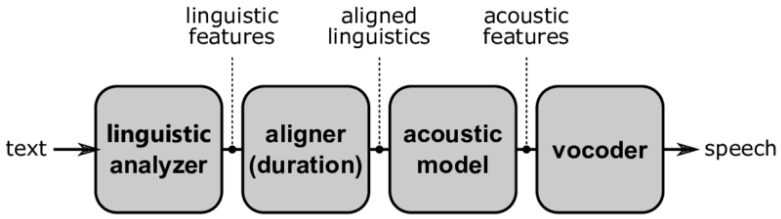

Figure 5. Conventional TTS pipeline representation

Once textual insight has been extracted, PsychExtract produces an accessible output for users. Shen et al. introduced Tacotron 2, a leading TTS model capable of generating highly natural-sounding audio using a sequence-to-sequence architecture and a neural vocoder [9]. Their work demonstrated that TTS systems can reliably convert text into expressive speech.

In PsychExtract, TTS is used not for interaction but as an accessibility feature. The system reads out the generated summaries, emotional indicators, and cognitive themes for users who prefer auditory feedback or have reading difficulties. This completes the pipeline by providing an intuitive and inclusive output format.

Together, the OCR-NLP-TTS structure forms the full workflow. Handwritten or typed text is digitised; emotional, cognitive, and linguistic markers of early insight are extracted; and results are returned as both text and spoken summaries.

DESIGN

OVERVIEW

PsychExtract is designed as a modular analysis system that transforms unstructured mental-health-related text into structured, inspectable outputs. Rather than operating as a monolithic model, the system is organised as a sequence of interoperable components, each responsible for a clearly defined stage of processing. This design choice reflects both the heterogeneous nature of mental-health documentation and the project's emphasis on transparency, interpretability, and comparative evaluation.

At a high level, the system ingests handwritten or typed documents and processes them through a pipeline consisting of OCR, NLP-based emotion classification, an interpretability layer, and optional TTS output. Each stage produces intermediate artefacts that can be inspected independently, enabling users to trace how raw input text is transformed into higher-level representations. This structure supports human-in-the-loop interaction, allowing users to review, validate, or disregard automated outputs as needed.

A key design objective is flexibility. For each component, multiple pre-trained models or techniques can be substituted and compared (for example, alternative OCR engines or transformer-based classifiers). This enables systematic benchmarking under consistent conditions and prevents early design decisions from constraining later experimentation. The system therefore prioritises configurability and controlled experimentation over end-user optimisation or deployment readiness.

The overall design follows Template 4.1 (Orchestrating AI Models to Achieve a Goal), which is well-suited to workflows that require coordination across multiple specialised models. This orchestration-based approach supports maintainability, clear separation of concerns, and future extensibility, allowing new models or analytical layers to be incorporated without restructuring the entire system. It also ensures that performance trade-offs can be evaluated at both component and pipeline levels.

This design section elaborates on the context in which the system operates (domain and users), the rationale behind each component, and the architectural structure that connects them. It further outlines the planned development timeline, feasibility considerations, and evaluation strategy that guide early prototyping and subsequent iterations. Together, these elements define the design space within which PsychExtract is explored and assessed.

DOMAIN AND USERS

Domain

PsychExtract operates within the domain of psychological text analysis, where language is subjective, emotionally nuanced, and shaped by context. Systems in this domain must balance clarity, interpretability, and accuracy. Clarity is essential because psychological language is often ambiguous and unclear outputs can distort users' understanding of emotional content [7]. Interpretability equally important due to practitioners and researchers consistently report preferring transparent models whose reasoning can be traced and scrutinised [14]. Accuracy remains important as it underpins the reliability of emotion classification, yet even widely used datasets such as GoEmotions demonstrate macro-F1 scores around 0.46, highlighting the inherent

difficulty of affective text interpretation [4]. These constraints shape PsychExtract's design toward explainability, transparent processing stages, and systematic comparative evaluation.

Users

Three primary user groups inform the system's requirements. These are psychotherapists, psychology students, and psychological researchers, each with distinct expectations.

Psychotherapists benefit from concise and transparent summaries that supplement (not replace) their judgment. Prior work shows that clinicians adopt AI tools more readily when uncertainty is visible and the model's reasoning is inspectable [15]. PsychExtract therefore prioritises explainability, editable intermediate results, and conservative output framing.

Psychology students require an accessible way to explore how linguistic patterns relate to emotional meaning. Interpretable model outputs and editable OCR text support their learning without requiring technical expertise [7].

Psychological researchers need reproducible, modular pipelines that expose preprocessing decisions, classification steps, and error analysis. The system's modular design supports methodological transparency and replicable experimentation, consistent with current expectations in explainable affective NLP [16].

Framing the system around these three groups ensures that PsychExtract remains interpretable, trustworthy, and aligned with real-world mental-health practice.

COMPONENTS

PsychExtract integrates OCR, transformer-based NLP, linguistic interpretability layers, and TTS to automate the transformation of raw notes into structured insights. Automating this transformation reduces administrative burden and frees mental-health professionals to focus on synthesis, interpretation, and clinical care; reflecting well-established benefits of semi-automated documentation systems [17, 18]. A comparative modelling approach is applied across all components.

OCR

OCR converts handwritten or scanned text into machine-readable form. Tesseract and EasyOCR evaluated comparatively. Tesseract is computationally lightweight and reliable for clean printed text [8, 19], while EasyOCR's deep-learning architecture handles noisy variable inputs more robustly [20]. Evaluating both determines which is better suited to real unstructured mental-health documentation.

Emotion Classification

Emotion classification facilitates affective insight central to psychosocial interpretation [2]. Two transformer models are assessed: DistilBERT, which is a more compact model aiding efficiency without significantly compromising its understanding capabilities [21], making it a strong candidate for lightweight deployment; and RoBERTa, which typically yields a higher accuracy due to optimised pretraining [22]. The systematic benchmarking of both models balances performance with computational feasibility.

Interpretability Layer

Linguistic interpretability accompanies extracted emotion predictions to offer transparent, psychologically meaningful explanations [23, 24]. KeyBERT offers embedding-driven keyword

extraction that captures semantic relevance [5], while alternative linguistic metrics (TF-IDF weighing or LIWC-style markers) offer rule-based, clinically explainable intuitive signals [23, 24]. Comparing these methods allows evaluation of semantic against rule-driven interpretability.

TTS

TTS provides multimodal accessibility for users with reading, attentional, or cognitive challenges [25]. Coqui TTS provides high-quality, neural speech synthesis [26], whereas pyttsx3 offers a reliable offline alternative suitable for constrained environments [27]. This comparison supports accessibility-oriented design, consistent with evidence that TTS enhances comprehension and engagement for diverse users [25].

Design Principles

The system follows core clinical-AI design guidelines [28]. The following principles are emphasised across the development of PsychExtract: modularity for safe isolation and replacement of models, transparency through visible intermediate layers, human-in-the-loop editing of OCR and review of generated insights, and minimal user interface (UI) for the purpose of emphasising clarity and explainability.

A functional UI implemented in Streamlit enables immediate testing. Figma supports early exploration of richer interface designs. This enables iterative refinement consistent with best practices in digital mental-health tool development [29].

ARCHITECTURE AND STRUCTURE

PsychExtract uses a modular, linear pipeline in which each stage is independently testable and replaceable. This maximises interpretability, facilitates comparative evaluation, and maintains clear traceability across the workflow. Data progresses through five main stages: OCR, emotion classification, linguistic interpretation extraction, summary generation, and optional TTS.

Pipeline Stages

1. Input Upload: user provides handwritten or scanned document
2. OCR: Tesseract/EasyOCR produces preliminary text
3. Editable Text: user verifies and corrects OCR output.
4. Emotion Classification: DistilBERT/RoBERTa produce multi-label probabilities
5. Interpretability Layer: KeyBERT or linguistic metrics extract explanatory features
6. Summary Generation: concise psychological insight produced
7. Optional TTS: pyttsx3/Coqui converts summary to speech
8. User Feedback Capture: supports iterative refinement and future evaluation

User Flow Diagram

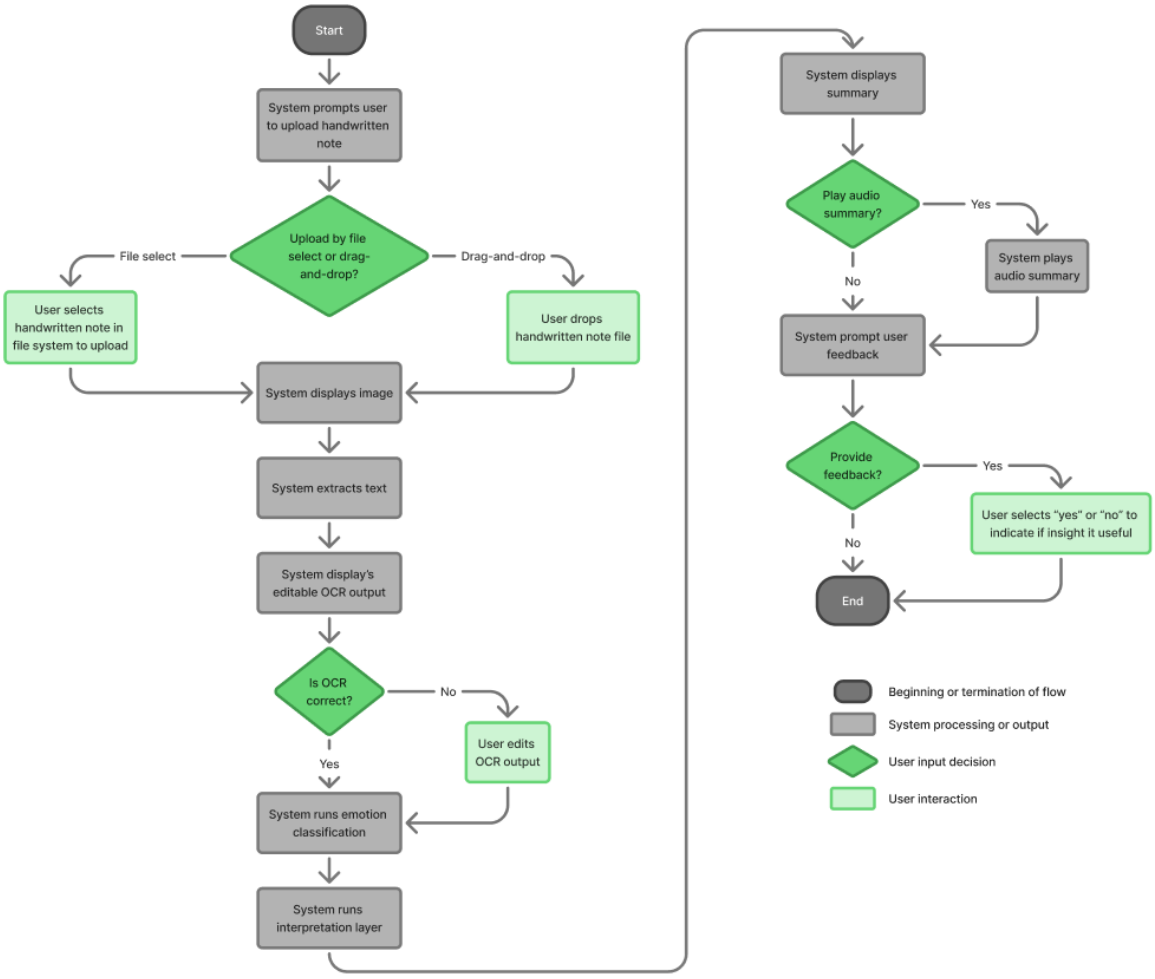

Figure 6. PsychExtract user flow

Early Prototype

A low-fidelity Streamlit UI demonstrates uploading, editing, and reviewing workflows

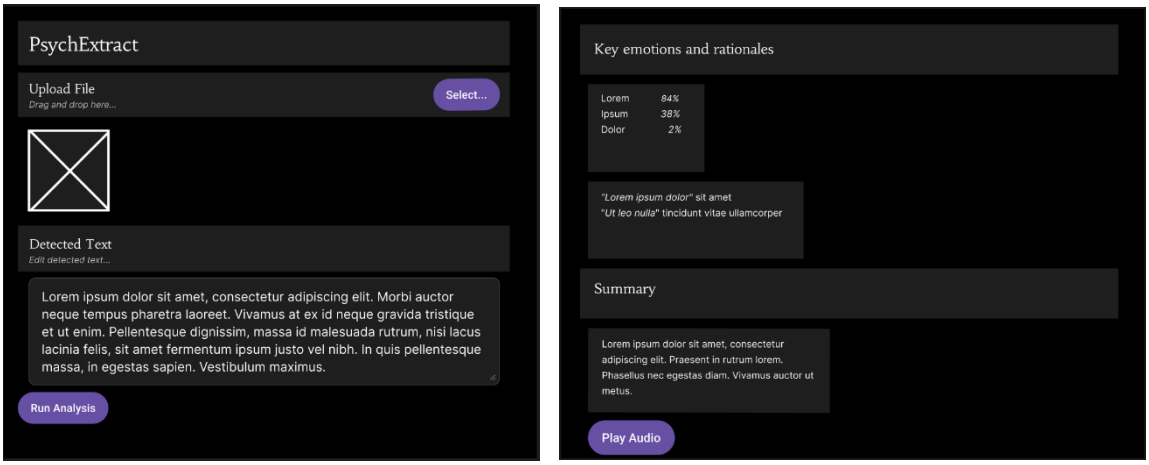

Figure 7. PsychExtract prototype interface

A more feature-rich annotator-style interface will be prototyped in Figma to explore alternatives without committing to full implementation.

Folder Structure

Folder	Purpose
ocr/	OCR wrappers for Tesseract and EasyOCR
nlp/	DistilBERT and RoBERTa models
interpret/	KeyBERT and linguistic metrics
summary/	Summarization and TTS components
ui/	Streamlit interface
tests/	Unit and integration tests

Table 1. PsychExtract folder structure

This modular organisation ensures traceability, flexible model swapping, and a clear separation of responsibilities between pipeline stages, facilitating efficient evaluation and iterative design.

WORK PLAN

The project runs from 8 December 2025 to 23 March 2026, following a structured iterative work plan divided across 4 months (see Figure 9). The workflow integrates continuous user testing and model evaluation.

Each major component (OCR, NLP, linguistic insights layer, TTS, and UI) follows a consistent cycle: a minimal working component is prototyped; it is evaluated through targeted tests (accuracy, robustness, performance, etc.); the component is subsequentially tested with users for clarity, usability, and interpretability; the component is then refined based on outcomes; finally, it is integrated into the overall pipeline. Model comparison occurs within each phase, with insights feeding informing iterative refinement.

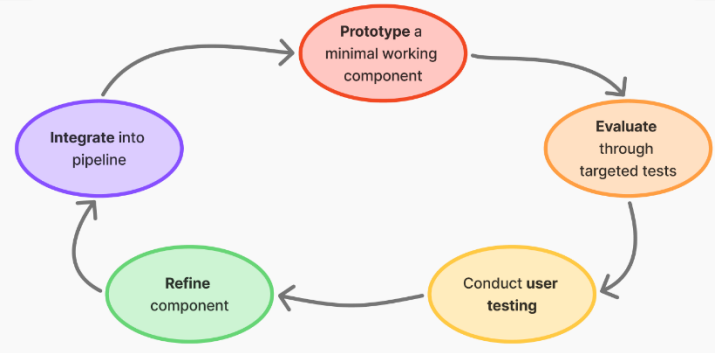

Figure 8. PsychExtract development cycle

Requirements and UI layouts are also tested early using low-fidelity prototypes, ensuring that design decisions reflect user needs before implementation. This continuous cycle of prototyping, evaluation, and refinement drives evidence-driven development and progressive system improvement from the initial concept to the final integrated system.

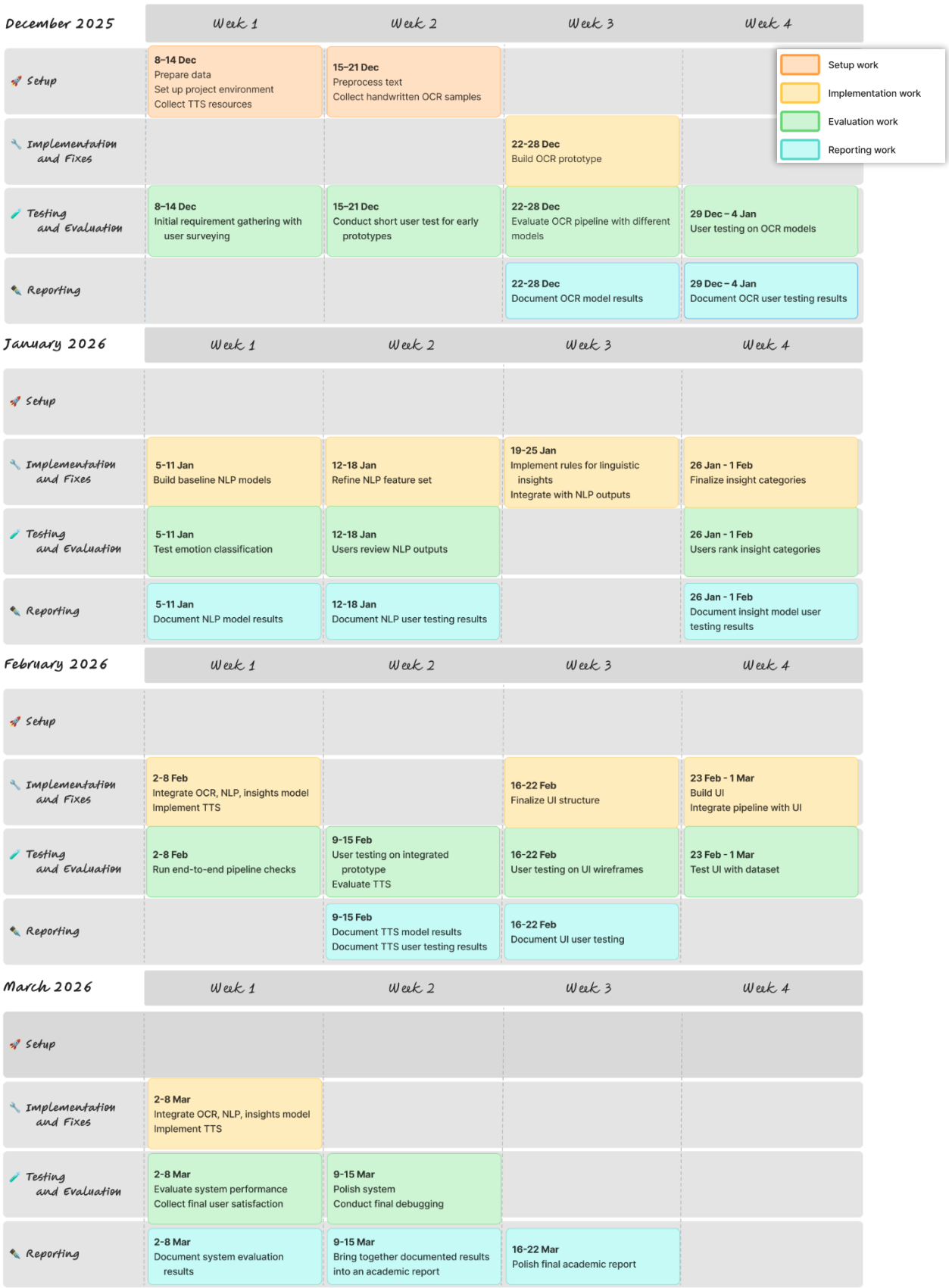

Figure 9. PsychExtract December 2025 to March 2026 work plan

FEASIBILITY AND CONTINGENCY PLANS

PsychExtract's design is realistic due to the modular architecture, lightweight model choices, and availability of both local and cloud-based compute. Transformer models used are compact enough for local execution, with Google Colab available as needed. The modular pipeline allows isolated debugging, clear performance benchmarking, and incremental development. The use of pre-trained models significantly reduces training cost and development time. Iterative testing embedded after each implementation stage enables early issue detection and prevents late-stage failures, supporting reliable progress toward final integration.

Contingencies strategies address component-level risks: If both pre-selected OCR engines perform poorly on highly degraded scans, input may be restricted to digital PDFs, or stronger pre-processing can be applied. For extremely poor-quality documents, a text-only workflow can serve as a fallback. If transformer-based NLP models exceed hardware limits, the TF-IDF logistic regression model (which is already part of the planned comparison) provides a fully functional alternative. If interpretability outputs prove insufficiently clear to users, a lexicon-based psychological category system ensures stable and clinically interpretable fallback explanations. For TTS, if neural synthesis causes latency or resource issues, the inclusion of pyttsx3 guarantees an offline, low-resource alternative. For the UI, if clinical users are unavailable for testing and validation, psychology students or general usability participants will act as representative testers, as critical tasks focus on clarity and interaction, rather than psychological expertise.

Together, the comparative design, iterative testing schedule, and targeted contingencies ensure robust, resilient, and feasible development towards final system integration.

EVALUATION PLANS

The evaluation strategy is aligned with PsychExtract's objectives: assessing model performance, interpretability, and user experience across all pipeline stages. Evaluation is embedded throughout development and repeated after full integration to ensure component-level insights translate into coherent system behaviour.

OCR Evaluation

OCR evaluation focuses on accuracy, robustness, and contextual error patterns. Character-level accuracy and word-error rate serve as core quantitative metrics, capturing both fine-grained recognition quality and overall textual coherence. Quantitative outcomes are supplemented by structured error analysis that identifies recurrent failure types (such as letter-shape confusions, mis-segmentation, and spacing considerations) offering insight into which OCR engine is better suited to variable real-world mental-health documentation.

NLP Classification Evaluation

Emotion classifiers are assessed using precision, recall, and macro-F1, which measure the quality of multi-label predictions, reflecting the uneven class distributions typical of affective datasets. Confusion matrices reveal systematic patterns of misclassifications, while qualitative error analysis examines mislabelled samples to determine whether model limitations, linguistic ambiguity, or upstream OCR noise. This dual analysis supports transparency by explaining how errors occur and why.

Interpretability Layer Evaluation

The interpretability layer is assessed through both quantitative coherence measures and qualitative judgement from target users. Topic coherence scores such as UMass and c_v assess whether

extracted key-phrases or clusters reflect meaningful semantic groupings [30]. Complementary readability indices, including Flesch–Kincaid [31] and lexical diversity metrics, quantify clarity and linguistic complexity in generated insights. User-testing oriented face-validity checks assess whether outputs appear plausible and psychologically aligned, ensuring interpretability remains grounded in real-world expectations. Integration checks verify that upstream errors do not propagate to distort downstream explanations.

TTS Evaluation
TTS performance considers both technical responsiveness and subjective usability. Latency (the time between text generation and audio output) indicates computational efficiency, while a small MOS-style naturalness rating [32] (where participants assess clarity on a scale of 1 to 5) captures perceived clarity and listening comfort.

Interaction and Usability Evaluation
Interaction and usability evaluation centres on transparency, trust, and ease of use. Cognitive walkthroughs examines whether new users can successfully upload documents, correct OCR results, understand extracted insights, and initiate TTS playback. A small number of users perform task-based usability tests (capturing behavioural observations, think-aloud reflections, and short post-task questionnaires) revealing friction points and guiding iterative refinement.

This evaluation plan ensures methodological rigour, supports comparative analysis across models, and maintains alignment with the system's aims of interpretability, transparency, and user-centred design.

PROTOTYPING

FEATURE MOTIVATION

The first prototype focuses on OCR because accurate transcription is the foundation upon which all downstream analyses in this system depend. The project relies on converting handwritten clinical notes into machine-readable text before any keyword extraction, sentiment analysis, or insight detection can occur. If the handwriting-to-text pipeline is unreliable, all subsequent components inherit those errors, amplifying noise and reducing interpretability. Early feasibility testing of OCR is therefore essential to determine whether the broader system is technically viable.

Handwritten clinical material presents well-documented challenges. Unlike printed text, handwriting varies widely in stroke shape, alignment, slant, and spacing, especially in therapeutic journals where notes are informal, expressive, and produced without standardisation. Prior work shows that handwriting is significantly harder for OCR models than printed text due to irregular glyph formation, inconsistent baselines, and overlapping characters, leading to elevated character-level and word-level error rates [8, 33]. Clinical notes also include domain-specific terminology, abbreviations, and idiosyncratic phrasing, which further complicate recognition.

By prototyping OCR early, the project can test model suitability, estimate expected error rates, and identify context-specific failure modes before investing in later stages of the pipeline. Establishing the reliability of text extraction at this point ensures that subsequent features (keyword extraction and emotional-insight modelling) are built on stable ground. This prototype thus serves as a critical checkpoint for feasibility, data quality, and system robustness.

To ensure that early experimentation did not involve sensitive or personal material, all handwritten inputs were created ethically. The textual extracts were sourced from anonymised public posts on r/offmychest, and each extract was manually rewritten by a volunteer to generate realistic handwriting samples. This approach avoids the use of private journals or identifiable personal data while still providing meaningful variability in handwriting. As such, the prototype operates on synthetic handwritten data that is representative but ethically safe.

PROTOTYPE DESCRIPTION

The prototype implements a lightweight OCR pipeline intended to assess the feasibility of recognising handwritten text and preparing it for downstream comparison. Its goal is not to optimise recognition accuracy, but to establish whether handwritten inputs can be reliably converted into machine-readable text and stabilised through basic preprocessing, forming the foundation for later similarity analysis.

The inputs consist of three short textual extracts originally sourced from anonymised public posts on r/offmychest, selected as examples of emotionally reflective writing. To generate realistic handwritten material without using private journals or personal documents, each extract was manually rewritten by a different volunteer. This resulted in three handwritten image samples, each exhibiting distinct handwriting characteristics. These images constitute the input set for the prototype, allowing the pipeline to be exercised on natural handwriting variation while remaining ethically safe.

Three OCR engines are incorporated into the prototype. Tesseract, a widely used classical OCR system [8], is included as a baseline due to its accessibility and prevalence in OCR applications. EasyOCR, a deep-learning-based recogniser with support for handwritten text [20], is included to represent a modern neural OCR approach that can be deployed with minimal configuration. In addition, PaddleOCR is integrated as a more advanced OCR framework designed to handle complex text layouts and handwriting more robustly [34]. Including all three engines enables the prototype to accommodate a range of OCR behaviours and ensures that downstream processing is not tied to a single recognition approach.

Processing follows a simple, sequential workflow. Each handwritten image is first loaded and passed independently through Tesseract, EasyOCR, and PaddleOCR, producing three separate textual outputs per image. These OCR outputs are then routed through a shared preprocessing stage that applies case-folding, punctuation normalisation, and whitespace cleanup. The resulting normalised texts are subsequently compared with the original ground-truth extracts to verify that the pipeline supports basic text comparison across different OCR sources.

The prototype is implemented in Python and executed in Google Colab, using standard image-handling utilities alongside the native APIs of each OCR engine. This environment enables rapid experimentation and supports both classical and deep-learning-based OCR systems without requiring extensive local configuration.

IMPLEMENTATION

```python
def paddleocr_ocr(image_files: list, ocr_engine: PaddleOCR) -> None:
    """
    Perform OCR on a list of image files using PaddleOCR and save the results.

    :param image_files: List of paths to the input image files.
    :type image_files: list
    :param ocr_engine: An instance of the PaddleOCR engine.
    :type ocr_engine: PaddleOCR
    """
    os.makedirs("paddle_pred", exist_ok=True)
    for img in image_files:
        print(f"Processing {img} with PaddleOCR")
        try:
            result = ocr_engine.predict(input=img)
            output_file = f"paddle_pred/output/{os.path.splitext(img)[0].split("/")[1]}"
            result[0].save_to_img(output_file)
            result[0].save_to_json(output_file)
            file_out = f"paddle_pred/{os.path.splitext(img)[0].split("/")[1]}_paddle_pred.txt"
            with open(f"{file_out}", "w") as f:
                f.write(" ".join(result[0]['rec_texts']))
            print(f"OCR complete for {img}. Result in {file_out}\n")
        except Exception as e:
            print(f"Error processing {img}: {e}\n")
```

Figure 10. Code snippet of PaddleOCR processing

The prototype was implemented as a single, sequential Google Colab notebook to ensure reproducibility and consistent processing across OCR engines. Three OCR libraries were integrated: Tesseract OCR, EasyOCR, and PaddleOCR, allowing comparative evaluation across different recognition approaches. Image handling and preprocessing were performed using Pillow (PIL), while jiwer, NumPy, and difflib supported quantitative and qualitative text comparison.

A unified preprocessing pipeline was implemented to ensure identical inputs across models. This included conversion to grayscale, contrast enhancement, and format normalization. JPEG inputs proved problematic due to compression artefacts degrading stroke clarity; images were therefore converted to PNG prior to OCR to reduce loss of fine handwriting detail. Preprocessing and evaluation logic were encapsulated into reusable functions to enforce architectural consistency and minimise experimental bias.

```python
def preprocess_images(image_paths: list) -> None:
    """
    Handle image preprocessing such as orientation correction and format conversion.

    :param image_paths: List of paths to the input image files.
    :type image_paths: list
    """
    for img in image_paths:
        im = Image.open(img)
        im = ImageOps.exif_transpose(im)
        im = im.convert("RGB")
        im.save(os.path.splitext(img)[0] + "_clean.png")
```

Figure 11. Code snippet of image preprocessing

Architecturally, the notebook follows a strict linear flow: image loading, preprocessing, OCR inference, text normalisation, evaluation. This design prioritises transparency and traceability over modular abstraction, which is appropriate for an early-stage prototype. OCR outputs are standardised to plain text to enable direct comparison using character-level metrics.

Several challenges were encountered. Dense handwritten pages caused partial text omission, likely due to overlapping strokes, inconsistent spacing, and low local contrast. Variations in lighting and page skew further impacted detection, occasionally leading models to ignore entire regions. These issues motivated conservative preprocessing choices to avoid over-filtering handwritten features.

EVALUATION

This section evaluates the feasibility of applying lightweight, off-the-shelf OCR engines (Tesseract, EasyOCR, and PaddleOCR) to handwritten, journal-like entries. The aim is not to optimise recognition accuracy, but to assess whether pretrained OCR systems can extract sufficient textual signal from unconstrained handwriting to support downstream natural language processing tasks. By comparing three widely used engines with different architectural assumptions, the evaluation identifies the practical limits of generic OCR for this domain.

Evaluation Method

Two standard OCR metrics were selected: character-level accuracy and word error rate (WER). Character accuracy provides a fine-grained measure of symbol-level correctness, while WER captures overall usability by quantifying substitutions, insertions, and deletions at the word level. These metrics are commonly used in OCR and handwriting recognition, particularly for evaluating performance on noisy or unconstrained inputs.

Three handwritten samples were collected from different individuals to reflect real-world variability in handwriting style. Ground truth texts were derived from r/offmychest excerpts. To ensure comparability, all OCR outputs and ground truth texts underwent identical preprocessing, including case folding, punctuation normalisation, and whitespace cleanup. Metrics were computed using custom character accuracy functions and the jiwer library for WER. Additionally, sequence similarity scores (via difflib.SequenceMatcher) were used as a supplementary indicator of overall textual overlap.

Quantitative Results

Engine	Character Accuracy	WER
Tesseract	0.067	0.829
EasyOCR	0.082	0.921
PaddleOCR	0.078	0.792

Table 2. OCR early prototyping character accuracy and WER

Across all engines, character accuracy remained below 10%, and WER exceeded 79%, indicating substantial recognition failure. PaddleOCR achieved the lowest average WER, while EasyOCR obtained the highest character accuracy, though with the poorest word-level performance. Sequence similarity scores mirrored these findings, with PaddleOCR producing markedly higher overlap on one sample (28% raw, 19% preprocessed), while all engines performed poorly on others.

Qualitative Error Analysis

Qualitative inspection revealed consistent error types across systems. Substitution errors were common, particularly involving visually similar character clusters (such as, “rn” vs. “m”, “cl” vs. “d”). Segmentation errors were prominent, with engines frequently splitting words incorrectly or merging adjacent words due to cursive connections. Omission errors occurred when strokes were faint or overlapped, resulting in dropped characters or entire words.

Tesseract produced the most severe segmentation failures, often generating character sequences that bore little resemblance to valid words. EasyOCR preserved some individual letter forms but frequently substituted short, high-frequency words incorrectly. PaddleOCR demonstrated more

stable line-level structure in some cases, explaining its lower WER, but still failed to maintain consistent sentence-level coherence.

Interpretation and Effect of Preprocessing

Overall, PaddleOCR exhibited the strongest relative performance, particularly in word-level error reduction, suggesting that its detection and recognition pipeline is more tolerant of handwriting variability. However, none of the engines achieved performance levels suitable for reliable downstream NLP. Preprocessing yielded marginal improvements in sequence similarity and character accuracy for some samples but did not fundamentally alter recognition outcomes. This indicates that preprocessing alone cannot compensate for the mismatch between pretrained OCR models (largely optimised for printed or structured text) and unconstrained personal handwriting.

From an early prototyping perspective, these results are informative rather than negative. They demonstrate that naïve application of generic OCR engines is insufficient for handwritten journal text, thereby providing clear empirical grounding for subsequent design decisions.

REFLECTIONS AND NEXT STEPS

This prototype demonstrated the feasibility of integrating pretrained OCR engines into PsychExtract while also exposing their clear limitations when applied to unconstrained handwritten journal entries. Although none of the evaluated engines achieved accuracy levels suitable for reliable downstream NLP, the comparative analysis provided valuable empirical insight into how generic OCR systems behave under realistic, noisy conditions. In particular, the results confirmed that off-the-shelf OCR can extract fragments of usable textual signal, but cannot be relied upon for faithful transcription of personal handwriting.

Among the evaluated engines, PaddleOCR will be favoured in the main system due to its relatively lower word error rate and more stable line-level segmentation. However, this preference reflects comparative robustness rather than adequacy; PaddleOCR remains insufficient as a standalone solution for handwritten input.

For the next development phase, these findings motivate a more modular OCR strategy. The system will retain multiple pretrained OCR engines to satisfy the requirement for heterogeneous data modalities, while allowing engine selection or fallback behaviour based on input characteristics (typed versus handwritten text). Given additional time, improvements would focus on enhanced image preprocessing, hybrid OCR pipelines combining multiple engines, and selective fine-tuning of pretrained handwriting-capable models. These steps would aim to improve recall while preserving transparency about OCR uncertainty in downstream analysis.

REFERENCES

[1] Clara E. Hill, Louis G. Castonguay, Lynne Angus, *et al.* 2007. Insight in psychotherapy: Definitions, processes, consequences, and research directions. In *Insight in psychotherapy*. Louis G. Castonguay and Clara E. Hill (Eds.), 441–454. American Psychological Association. https://doi.org/10.1037/11532-021

[2] Leslie S. Greenberg and Antonio Pascual-Leone. 2006. Emotion in psychotherapy: A practice-friendly research review. *Journal of Clinical Psychology* 62, 5 (2007), 611–630. https://doi.org/10.1002/jclp.20252

[3] David DeVault, Ron Artstein, Grace Benn, *et al.* 2014. SimSensei Kiosk: A Virtual Human Interviewer for Healthcare Decision Support. In *Proceedings of AAMAS '14* (May. 2014), 1061–1068. https://dl.acm.org/doi/10.5555/2615731.2617415

[4] Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, *et al.* 2020. GoEmotions: A Dataset of Fine-Grained Emotions. In *Proceedings of ACL* (Jul. 2020), 4040–4054. https://doi.org/10.18653/v1/2020.acl-main.372

[5] Maarten Grootendorst. 2025. KeyBERT. *GitHub*. Retrieved Nov. 26 2025 from https://github.com/MaartenGr/KeyBERT, *archived at* [https://web.archive.org/web/20250927082041/https://github.com/MaartenGr/KeyBERT]

[6] Maarten Grootendorst. 2022. BERTopic: Neural Topic Modelling with Transformers and class-based TF-IDF. *Manuscript submitted for review*. https://doi.org/10.48550/arXiv.2203.05794 archived at https://web.archive.org/web/20250808005533/https://arxiv.org/abs/2203.05794

[7] James W. Pennebaker, Ryan L. Boyd, Kayla Jordan, *et al.* 2015. The Development and Psychometric Properties of LIWC2015. *University of Texas at Austin*. https://doi.org/10.13140/RG.2.2.23890.43205

[8] Ray W. Smith. 2007. An Overview of the Tesseract OCR Engine. *In Proceedings of ICDAR '07 Vol. 2* (Sep. 2007), 629–633. https://doi.org/10.1109/ICDAR.2007.4376991

[9] Jonathan Shen, Ruoming Pang, Ron J. Weiss, *et al.* 2018. Natural TTS Synthesis by Conditioning Wavenet on Mel Spectrogram Predictions. In *Proceedings of ICASSP '18*, Apr, 2018, 4779–4783. https://doi.org/10.1109/ICASSP.2018.8461368

[10] Theresa A. Koleck, Caitlin Dreisbach, Philip E. Bourne, *et al.* 2019. Natural language processing of symptoms documented in free-text narratives of electronic health records: a systematic review. *Journal of the American Medical Informatics Association* 24, 4 (Apr. 2019), 364-379. https://doi.org/10.1093/jamia/ocy173

[11] Sankha S. Mukherjee, Jiawei Yu, Yida Won, *et al.* 2020. Natural language processing-based quantification of the mental state of psychiatric patients. *Computational Psychiatry* 5 (Dec. 2020), 76-106. https://doi.org/10.1162/cpsy_a_00030

[12] Rosanne J. Turner, Femke Coenen, Femke Roelofs, *et al.* 2022. Information extraction from free text for aiding transdiagnostic psychiatry: constructing NLP pipelines tailored to clinicians' needs. *BMC Psychiatry* 22, Article 407 (Jun. 2022), 11 pages. https://doi.org/10.1186/s12888-022-04058-z

[13] Jessica P. Ridgway, Amo Uvin, Jessica Schmitt, *et al.* 2021. Natural language processing of clinical notes to identify mental illness and substance use among people living with HIV: retrospective cohort study. *JMIR Medical Informatics* 9, 3, Article e23456 (Mar. 2021), 10 pages. https://doi.org/10.2196/23456

[14] Finale Doshi-Velez and Been Kim. 2017. Towards a Rigorous Science of Interpretable Machine Learning. *arXiv preprint*. https://doi.org/10.48550/arxiv.1702.08608

[15] Maia Jacobs, Jeffrey He, Melanie Pradier, *et al.* 2021. Designing AI for Trust and Collaboration in Time-Constrained Medical Decisions: A Sociotechnical Lens. *In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery 659, 1-14. https://doi.org/10.1145/3411764.3445385

[16] Saif M. Mohammad. 2022. Ethics Sheet for Automatic Emotion Recognition and Sentiment Analysis. *Computational Linguistics* 48, 2 (Jun. 2022), 239-278. https://doi.org/10.1162/coli_a_00433

[17] Dror Ben-Zeev, Kristin E. Davis, Susan Kaiser, *et al.* 2013. Mobile technologies among people with serious mental illness: opportunities for future services. *Administration and Policy in Mental Health and Mental Health Services Research* 40, 4 (Jul. 2023), 340-343. https://doi.org/10.1007/s10488-012-0424-x

[18] Tait D. Shanafelt, Lotte N. Dyrbye, Christine Sinsky, *et al.* 2016. Relationship between clerical burden and characteristics of the electronic environment with physician burnout and professional satisfaction. *In Mayo Clinic Proceedings* 91, 7 (Jul. 2016), 836-848. https://doi.org/10.1016/j.mayocp.2016.05.007

[19] Chirag I. Patel, Atul Patel, and Dharmendra Patel. 2012. Optical Character Recognition by Open Source OCR Tool Tesseract: A Case Study. *International Journal of Computer Applications* 55, 10 (Oct. 2012), 50-56. https://doi.org/10.5120/8794-2784

[20] JaidedAI. 2024. EasyOCR: Ready-to-use OCR with 80+ supported languages and all popular writing scripts including: Latin, Chinese, Arabic, Devanagari, Cyrillic, etc. *GitHub*. Retrieved Nov. 25 2025 from https://github.com/JaidedAI/EasyOCR, *archived at* [https://web.archive.org/web/20251012081255/https://github.com/JaidedAI/EasyOCR]

[21] Victor Sanh, Lysandre Debut, Julien Chaumond, *et al.* 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint*. https://doi.org/10.48550/arXiv.1910.01108

[22] Liu Yinhan, Myle Ott, Naman Goyal, *et al.* 2019. Roberta: A robustly optimized BERT pretraining approach. *arXiv preprint*. https://doi.org/10.48550/arXiv.1907.11692

[23] Jacob Devlin, Ming-Wei Chang, Kenton Lee, *et al.* 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* 1 (Jun. 2019), 4171-4186. https://doi.org/10.18653/v1/N19-1423

[24] Becky Inkster, Ross O'Brien, Emma Selby, *et al.* 2020. Digital Health Management During and Beyond the COVID-19 Pandemic: Opportunities, Barriers, and Recommendations. *JMIR mental health* 7, 7, Article e19246 (Jul. 2020), 5 pages. https://doi.org/10.2196/19246

[25] Mary Cece Young, Carrie Anna Courtad, Karen H Douglas, *et al.* 2018. The Effects of Text-to-Speech on Reading Outcomes for Secondary Students With Learning Disabilities. *Journal of Special Education* Technology 24, 2 (Jul. 2018), 80-91. https://doi.org/10.1177/0162643418786047

[26] Coqui.ai. 2025. Coqui.ai TTS. *TTS 0.22.0 documentation*. Retrieved Nov. 26 from https://docs.coqui.ai/en/latest, *archived at* [https://web.archive.org/web/20250812191341/https://docs.coqui.ai/en/latest]

[27] pyttsx3.readthedocs.io. 2025. pyttsx3 - Text-to-speech x-platform. *pyttsx3 2.6 documentation*. Retrieved Nov. 26 2025 from https://pyttsx3.readthedocs.io/en/latest, *archived at* [https://web.archive.org/web/20251010005655/https://pyttsx3.readthedocs.io/en/latest]

[28] Medicines and Healthcare Products Regulatory Agency (MHRA). 2025. Software and artificial intelligence (AI) as a medical device. *gov.uk*. Retrieved Nov 27. 2025 from

https://www.gov.uk/government/publications/software-and-artificial-intelligence-ai-as-a-medical-device/software-and-artificial-intelligence-ai-as-a-medical-device, *archived at* [https://web.archive.org/web/20250815022341/https://www.gov.uk/government/publications/software-and-artificial-intelligence-ai-as-a-medical-device/software-and-artificial-intelligence-ai-as-a-medical-device]

[29] John Torous, Jennifer Nicholas, Mark E. Larsen, *et al.* 2018. Clinical review of user engagement with mental health smartphone apps: evidence, theory and improvements. *Evidence-based mental health* 21, 3 (Aug. 2018), 116-119. https://doi.org/10.1136/eb-2018-102891

[30] Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the Space of Topic Coherence Measures. *In Proceedings of the Eighth ACM International Conference on Web Search and Data Mining (WSDM '15)* (Feb. 2015), 399-408. https://doi.org/10.1145/2684822.2685324

[31] Peter J. Kincaid, Robert P. Fishburne Jr., Richard L. Rogers, *et al.* 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. *Technical Report Research Branch Report, Millington, TN: Naval Technical Training*. Article RBR875, 51 pages. https://doi.org/10.21236/ADA006655

[32] ITU-T P.800. 1996. Methods for subjective determination of transmission quality. *International Telecommunication Union*. 37 pages.

[33] Réjean Plamondon and Sargur N. Srihari. 2000. On-line and off-line handwriting recognition: a comprehensive survey. IEEE Trans Pattern Anal Mach Intell (T-PAMI). *IEEE transactions on pattern analysis and machine intelligence* 22, 1, 63-84. https://doi.org/10.1109/34.824821

[34] Cheng Cui, Ting Sun, Suyin Liang. 2025. PaddleOCR-VL: Boosting Multilingual Document Parsing via a 0.9B Ultra-Compact Vision-Language Model. *arXiv preprint*. https://doi.org/10.48550/arXiv.2510.14528