

Article

HIPPP: Health Information Portal for Patients and Public

Colm Brandon ^{1,*} , Adam J. Doherty ² , Dervla Kelly ^{3,4,5} , Desmond Leddin ^{5,6} and Tiziana Margaria ^{1,2,3,5} ¹ Centre for Research Training in Artificial Intelligence (CRT AI), T12 XF62 Cork, Ireland; tiziana.margaria@ul.ie² Lero-Science Foundation Ireland Research Centre for Software, University of Limerick, V94 T9PX Limerick, Ireland³ The Health Research Institute (HRI), University of Limerick, V94 T9PX Limerick, Ireland; dervla.m.kelly@ul.ie⁴ Faculty of Education and Health Sciences, School of Medicine, University of Limerick, V94 T9PX Limerick, Ireland⁵ University of Limerick Cancer Network (ULCaN), University of Limerick, V94 T9PX Limerick, Ireland⁶ Division of Digestive Care & Endoscopy, Department of Medicine, Dalhousie University, Halifax, NS B3H 4R2, Canada

* Correspondence: colm.brandon@ul.ie

Abstract: Cancer misinformation is becoming an increasingly complex issue. When a person or a loved one receives a diagnosis of possible cancer, that person, family and friends will try to better inform themselves in this area of healthcare. Like most people, they will turn to their clinician for guidance and the internet to better verse themselves on the topic. But can they trust the information provided online? Are there ways to provide a quick evaluation of such information in order to prevent low-quality information and potentially dangerous consequences of trusting it? In the context of the UL Cancer Research Network (ULCaN), this interdisciplinary project aims to develop the Health Information Portal for Patients and Public (HIPPP), a web-based application co-designed with healthcare domain experts that helps to improve people navigate the health information space online. HIPPP will be used by patients and the general public to evaluate user-provided web-based health information (WBHI) sources with respect to the QUEST framework and return a quality score for the information sources. As a web application, HIPPP is developed with modern extreme model-driven development (XMDD) technologies in order to make it easily adaptable and evolvable. To facilitate the automated evaluation of WBHI, HIPPP embeds an artificial intelligence (AI) pipeline developed following model-driven engineering principles. Through co-design with health domain experts and following model-driven engineering principles, we have extended the Domain Integrated Modelling Environment (DIME) to include a graphical domain-specific language (GDSDL) for developing websites for evaluating WBHI. This GDSDL allows for greater participation from stakeholders in the development process of both the user-facing website and the AI-driven evaluation pipeline through encoding concepts familiar to those stakeholders within the modelling language. The time efficiency study conducted as part of this research found that the HIPPP evaluation pipeline evaluates a sample of WBHI with respect to the QUEST framework up to 98.79% faster when compared to the time taken by a human expert evaluator.

Keywords: model driven engineering; artificial intelligence; machine learning; WBHI; health informatics; web-based health information; DIME



Citation: Brandon, C.; Doherty, A.J.; Kelly, D.; Leddin, D.; Margaria, T. HIPPP: Health Information Portal for Patients and Public. *Appl. Sci.* **2023**, *13*, 9453. <https://doi.org/10.3390/app13169453>

Academic Editor: Anton Civit

Received: 20 June 2023

Revised: 3 August 2023

Accepted: 7 August 2023

Published: 21 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In the pre-internet era, individuals typically acquired information through direct interaction with what could be defined as reliable sources: experts, teachers, professionals, or printed media (encyclopedias, newspapers, etc.), which were subject to content supervision/vetting and strict publishing standards. In the online era, the nature of how people obtain information has changed significantly, moving towards indirect, ad hoc connections with a multiplicity of sources that may have low levels of reliability [1].

As of 2019, 91% of Irish households had internet access and 63% of Irish internet users used the internet to access general healthcare information [2]. A 2011 study found that 63% of cancer patients use the internet to find healthcare information [3]. Given the general increase in internet adoption since then, it is likely that the current figures are even higher.

For individuals with a non-medical education, such as most cancer patients and their loved ones, it is difficult to judge the reliability of WBHI. In the circumstance of a cancer diagnosis, it is natural that such individuals may turn to the internet to better educate themselves, verify the information they received from their physician, develop questions to discuss with their physician and/or seek alternative treatments. Given that inaccurate or incomplete healthcare information is so prevalent, the current information space is difficult to navigate. Considering that a 2014 study found that “improvement of knowledge obtained through personal research on the internet, books and other media” is an independent predictor of an active role in the choice of therapy [4], the current WBHI landscape may be causing some patients to base their healthcare decisions on incomplete or inaccurate information, with a detrimental effect on their health outcomes.

The need for evaluation of WBHI quality was already recognised in the early days of the internet and a number of tools have been developed in the intervening years. The most notable evaluation tools are the JAMA criteria [5] and DISCERN [6]. Another instrument of note is The Health On the Net Code of Conduct (HONcode) for medical and health websites [7]: websites may display the HONcode seal if they agree to comply with their listed standards and are subjected to random audits to ensure compliance. However, there are significant issues with these existing methods for combating low-quality WBHI. JAMA and DISCERN are both manual: an expert must examine a website and evaluate it against the scoring criteria outlined in the respective framework. Given the volume of WBHI information currently on the internet and the rate at which new WBHI is being published, manual and time-consuming processes are not feasible because they do not scale and they are prone to individual interpretation of the criteria, violating uniformity. With respect to the HONcode instrument, a 2020 study about COVID-19 misinformation found that less than 2% of the websites returned by their Google searches had the HONcode seal [8]: HONcode is de facto irrelevant for the majority of internet users.

Given both the growing but fragmented WBHI landscape and the current deficiencies in tools to combat low-quality WBHI, our project within the UL Cancer Network (ULCan) (<https://www.ul.ie/hri/university-limerick-cancer-network-ulcan> (accessed on 1 June 2023)) PPI Pillar aims to develop a web application that efficiently automates the evaluation of WBHI sources with respect to a number of criteria adopted from the established WBHI evaluation framework QUEST [9]. QUEST is designed to reduce the need for subjective judgement calls when compared to similar tools. QUEST has undergone a validation process and was bench-marked against the DISCERN and HONcode tools. Whilst the QUEST tool is considerably quicker to apply than similar tools, there are still issues with the length of time that patients or physicians take to apply it to each website they encounter. Similarly, the average patient may lack the skill set to critically evaluate sources, conflicts of interest, etc. To address this, we develop an AI pipeline to automate the application of QUEST to WBHI. While the scope of the HIPPP project is restricted to the domain of colon cancer information, because we have within the Patient and Public Involvement (PPI) initiative a network of participating patients and experts in this specific health domain (who will provide feedback throughout the development process), the developed system is general purpose and easily adaptable to other kinds of cancer and other domains of health. The application will be available for use primarily by patients and the general public in the Republic of Ireland. In terms of adopted technologies, the web application is developed sustainably using XMDD technologies, and it embeds an AI pipeline for the evaluation of sites that are the sources of health information.

Our contributions are as follows:

- With respect to *domain-specific languages for health and medical research support*, our contribution is the extension of the DIME platform through the creation of a GDSL

and integration of the accompanying set of services for that GDSDL, which will in future enable health and medical researchers to develop their own equivalent web application or be much more integrated into the design and development process.

- Pertaining to *data management cycle for integration and longer reuse*, the HIPPP user-facing application will serve as a crowd-sourcing tool. With the acquired data (i.e., the user-provided web-pages) being stored to support future uses. The primary use being to improve the applications' performance through human-in-the-loop incremental learning, however, it could also serve the function of giving researchers insight into what health information sources patients and the public use.
- In the context of *technologies for the evolution and customisation of IT platforms for health*, due to the co-design element of the HIPPP project, we were able to encapsulate concepts familiar to those working in the healthcare and medical domain to customise the web application to meet the requirements and expectations of medical domain researchers.

In the following, Section 2 presents the web-based application and discusses its features. In Section 3, we discuss the methodologies used in the development of the HIPPP user-facing application, the evaluation pipeline and the incremental learning [10] component of the overall application. We also discuss the different approaches taken to the validation of both the web-based application and the evaluation pipeline. As these two aspects of the application tackle different goals, in the validation, we take different approaches to the verification process, as most adequate for the specific goal. Finally, Section 5 will outline the results to date and introduce the next steps in the project in the health informatics domain.

2. Materials: The HIPPP IT System for PPI

In this section, we describe the two main components of the HIPPP IT system: the web-based application and the AI pipeline for the information evaluation that is embedded in the HIPPP web application.

2.1. The HIPPP Web-Based Application

The HIPPP system is implemented as a web-based application whose front end contains the entry point for the WBHI evaluation pipeline.

The aim of this application is to provide an intuitive and appealing user interface (UI) for cancer patients and the general public so that they can access the automated quality evaluation for cancer-specific WBHI. The expected user group of the HIPPP application is the general public; therefore, the design of the UI is crucial to its success. We follow the criteria of simplicity and a welcoming style for the UI in order to make it easy to use and also pleasant to use and welcoming by, e.g., adopting colours that the users find attractive and associate with positive feelings.

In order to make it easily adaptable, it is designed following the XMDD [11] approach and uses a technology that supports the easy evolution of both the user interface and the workflows and processes in the application.

The HIPPP UI has been designed following best practice Human–Computer Interaction (HCI) principles, to aid in the inclusion of the expected user groups [12]. Taking into account the expected user group, the application is in fact required to be *accessible* to all users with different accessibility needs. The HCI principle of accessibility addresses discriminatory aspects related to ensuring an equivalent user experience for people with disabilities [13] by the Web Accessibility Initiative (WAI).

Simplicity is a core principle of the UI design: when users find an online resource that contains cancer-based information, they can have its quality evaluated by the HIPPP application by submitting the URL to HIPPP. Accordingly, the main functionality of the application is to allow users to input a URL in a very simple form and send it to the HIPPP evaluation pipeline for automated quality evaluation. The output of this scoring process is then returned to the user: it shows the outcome of the evaluation and provides a breakdown

of the different heuristics on which it has been scored. This output enables a user to make a better-informed decision on whether the contents of a given website should be trusted. Figure 1 shows a view of the search page of the HIPPP application.

The HIPPP application also includes a *curated list*. The purpose of the curated list is to provide users attempting to gather more information about their or loved ones' diagnosis with a list of pre-evaluated websites so that users immediately find a selection of sites that should be trusted. It also includes a list of websites that should not be trusted to inform users that the information contained on these websites should not be trusted, as it may be of low quality. These lists are provided and curated by domain experts in the specific field of cancer research. As shown in Figure 2, for us this is initially colon cancer. The aim of this functionality is to provide users with references for reputable websites when searching for WBHL.



Figure 1. The HIPPP landing page.



Figure 2. A view of the web page, which displays the curated sources list. This list is curated by cancer domain experts and includes trusted sources (green), which users are encouraged to seek information on, and sources that are not trusted (red), which users are encouraged to avoid.

A further contribution of the HIPPP application is to host a *labelling system* for users of the portal that are subject area experts (like colon cancer surgeons, nurses, radiologists, nutrition experts and dietitians, etc.) who provide a manual classification of WBHI to aid in the HIPPP classifier incremental learning process. The details of this feature are outlined in Section 2.2.7, but its inclusion in the HIPPP web application is due to the requirement of including expert human augmentation and checks. In a scenario where the HIPPP pipeline is unable to provide an evaluation and thus a recommendation with sufficiently high confidence, a view of the source and the pipeline evaluation outcome is displayed to expert users, who have a separate role in the application. Such queries are queued in order to be reviewed by appropriate domain experts. This feature will allow expert users to log in to the HIPPP application in an expert user role and review the contents of queries that have been collected for expert referral. The experts can see the content of the website in question, and in multiple views, they use the provided labelling system to label the fragments of WBHI content that led the automatic evaluation pipeline to a low confidence rating. The content is presented in multiple views in order to allow the expert user to label the content for training different machine learning models. Once an expert user is finished with labelling the content, both the content and the associated labels are then sent to the incremental learning portion of the evaluation pipeline. This way, over time, the system's ability to automatically evaluate WBHI improves.

2.2. HIPPP-AI: The Evaluation Pipeline Environment

The evaluation pipeline is the automated system that takes the input URL provided by users via the front-end application and processes and rates the WBHI content of the corresponding website. Specifically, as shown in Figure 3, it scrapes the website, analyses its content, performs a content classification with respect to the QUEST framework, calculates a confidence value that the classification is correct and, based on the outcome of the confidence analysis, it either presents the result to the user or in case of low confidence, it refers the given WBHI to the expert evaluation process. We now present the individual phases of the workflow.

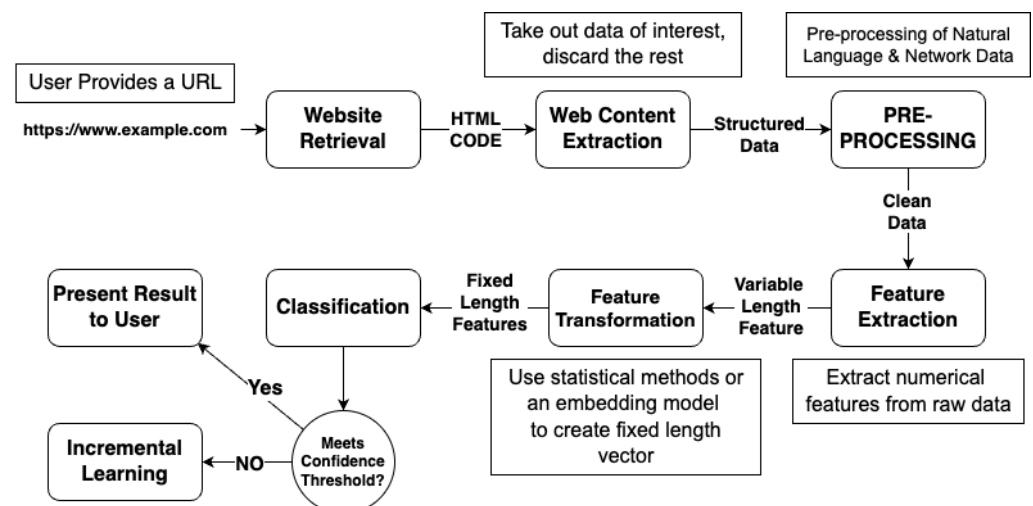


Figure 3. Overview of the HIPPP evaluation pipeline.

2.2.1. Website Retrieval

This process retrieves and saves the raw HTML code that the given URL points to. As the URL is user-provided, some checks ensure it is a valid URL and that the HTTP response to the server returns a valid response and a valid status code. Given that studies show that approximately 1 in 14 web servers will return a “soft-404” code [14], we have also implemented checks for this, performed on the retrieved HTML code.

2.2.2. Web Content Extraction

A HTML document such as a web page can be viewed as either unstructured or semi-structured data. The web content extraction process analyses the HTML document, keeps what is wanted from this unstructured data, stores it in a structured format and discards the rest. For instance, our analysis is not concerned with the navigation menu nor the footer, which can be discarded, and it is concerned with the article body and the author section. The natural language text and network data are extracted from the elements of concern, and some global elements such as metadata are also extracted.

2.2.3. Data Pre-Processing

AI algorithms tend to perform better on data when it is stored in a specific format or require data in a specific format to simply work; therefore, it is important to pre-process the input data. Their analysis uses two main types of data: the natural language text and the network data. The network data come in the form of a set of URLs, which are subdivided into out-links and back-links. Using a series of pre-processing steps, this network of links is converted to a directed graph with a set of nodes (the web pages) and a set of directed edges (the out-links and back-links) to other pages. Each node contains numerical features describing the node, which is the required input format for a series of network analysis algorithms. Similarly, with natural language, pre-processing steps on the text are needed. An example is lowercase homogenisation, which makes all the letters lowercase. “Hello” and “hello” would be otherwise unnecessarily perceived as being two completely independent entities, therefore making the feature space twice as big and, consequently, some problems harder to solve and analyses less efficient. However, for some tasks such as named-entity recognition (finding a person’s or company name in some text), capital letters are an important semantic feature, so removing them all would make this problem harder to solve. Therefore, the type of pre-processing applied at this stage is very much informed by the requirements of the downstream tasks.

2.2.4. Feature Extraction

Performing the evaluation with respect to the QUEST framework and the seven criteria in QUEST requires certain features to be extracted from a given piece of WBHI. Some features were manually designed in order to leverage existing machine learning algorithms to extract them, but in the majority of cases, we had to develop some novel feature extraction methods. For example, when determining the *tone* of a WBHI resource for the QUEST criteria, taking the set of pre-processed sentences (natural language) in the article as inputs, a tone detection algorithm will classify each sentence as expressing a “for”, “against” or “neutral” tone. The results of this process are the set of classification results for the set of sentences. This can then be viewed as a feature vector that describes the tone of the whole article.

2.2.5. Feature Transformation/Embedding

Machine learning and rule-based classifiers require their input vectors to be of a predefined fixed length n . Considering again the sentence tone example, in an article, there can be an arbitrary number of sentences, which conflicts with the fixed length vector requirement for classifiers. To resolve this, the feature transformation process takes the extracted features and maps them to that fixed length. This is achieved through statistical methods, dimensionality reduction or using neural networks to create embeddings.

2.2.6. Classification

The classification proper comprises of an ensemble of classifiers, where each classifier pertains to a specific part of the QUEST criteria: authorship, attribution, type of study, conflicts of interest, currency, complementary and tone. Each criterion-specific classifier takes as input its relevant fixed-length feature vectors and assigns a QUEST score.

The final step is confidence analysis: while each classifier is rule-based and therefore contains no notion of confidence, the models used in the feature extraction phase do. The confidence of each of the performed classifications is analysed. If the cumulative confidence meets a certain threshold, the labels are presented to the user via the front-end application. If the cumulative confidence is below the threshold, the WBHI is referred to the expert evaluation. Once it is reviewed by the experts, the corresponding result will be sent to the user.

2.2.7. Incremental Learning

Several studies created corpora of WBHI and evaluated the quality of the contained information through one of the QUEST, HONCode or DISCERN frameworks: 269 samples were considered in [15], 200 samples in [16], and 35 samples in [17]. However, these are tiny sample sets, leading to insufficiently small corpora. These data sets are not comprehensive enough to create truly generalised and robust models.

Given this scarcity of available suitable corpora, we explored the idea of creating such an exhaustive data set for this task. The first task was to identify “gold standard” information sources. We consider government entities (such as hse.ie) and advocacy groups (such as cancer.ie) as reliable and relatively straightforward to find via search engine queries. However, aside from those most obvious content providers, determining where else to look in order to find real examples of the kinds of information sources that people engage with is a non-trivial exercise. The reason is the complexity of the information propagation channels on the internet. On top of this, there is the social media layer, where the information served to an individual is personalised based on a plethora of factors unknown to anyone just using or accessing social media.

In this context, it would be extremely difficult, if not infeasible, for a small group of researchers to source and gather a collection of samples of sufficient quantity and representative quality to train high-quality models before making the pipeline accessible to the public. As a result of this initial data scarcity, the strategy was devised to use the HIPPP system itself to serve as a crowd-sourcing tool for gathering real-world samples of WBHI that real portal users do access on the internet. So, along with providing the user with an evaluation of the user input samples, the HIPPP will also serve as a data collection tool for information sources that are then labelled either automatically or manually through the expert portal.

This approach enables *incremental learning* through the process of adding new training data over time to extend the model’s knowledge while retaining the knowledge learned from existing data. Given that we have access to a network of physicians who specialise in cancer research, we can leverage their domain knowledge to label “hard samples”, i.e., those which did not meet the confidence threshold described above. The labelled data can then be used to extend the knowledge of the various models in the pipeline and to improve their performance.

It is important to note that many of the tasks undertaken in the pipeline, such as coreference resolution and tone detection, are well-established research areas and therefore data scarcity is not an issue. Although available training data for these tasks is not specific to the domain of medical information, transfer learning [18] can be undertaken with a small volume of domain-specific data to tune the models.

3. Methods

In this section, we describe the development methodology used in the creation of the HIPPP web-based application, the evaluation pipeline components and the incremental learning system. Firstly, we describe in Section 3.1 the model-driven development process of the public-facing web application and its design. We then describe the approach to the development of the evaluation pipelines through a model-driven approach in Section 3.2, and finally in Section 3.2.2 we discuss the incremental learning process that is used to further refine the pipeline’s accuracy.

3.1. Model Driven Development

In the development of the HIPPP public-facing portal, which we see also as the UI for the evaluation pipeline, we have adopted a model-driven approach. To create the portal, we use DIME [19], a modern low-code model-driven application development environment targeted to the continuous design, generation and deployment of web applications. Developing applications through DIME does not follow the traditional textual development convention and it is instead conducted through a graphical modelling environment. This approach to development enables the rapid development of prototypes for applications. DIME applications are deployed and orchestrated through Docker [20], an enterprise-standard virtualisation platform for application orchestration and deployment. This allows us to abstract the requirements for the application away from the infrastructure where it will be deployed and focus entirely on the application's development. Using this approach and DIME facilitates rapid application prototyping and development while still upholding web application development best practices. The methodology of modelling software in this way has multiple benefits to the person developing the software. These models are used to describe the data, the UI, and the workflows or business logic, i.e., what the application should be doing at any point in its working. Taking this approach together with a service-orientated view of each function has also the effect of minimising the (re-)development of similar components by abstracting the requirements of each piece of logic and then developing them in a generalised way. By introducing a uniform abstract model of these components (similar to a component management language), they can then be reused throughout the application and also in future projects across different domains.

We now describe how we leveraged DIME and the model-driven approach in the development of the front-end application.

DIME organises the development of any application through the following three distinct, aspect-specific and automatically interlinked model types.

3.1.1. Graphical User Interface (GUI) Modelling

The GUI models in DIME are used to define both the look and the functions of the UI for an application. Their almost “what you see is what you get” fashion enables the rapid prototyping of GUI models [21]. While developing these models, we took care of enhancing their reusability both in different locations throughout the application and also in other applications. Rapid prototyping speeds up the creation of multiple different prototypes for the UI, which becomes a much faster process than traditional textual UI development. This agility allows for multiple different iterations of the application's design and therefore enables a more refined design, as multiple options can be presented to stakeholders.

In the initial stages, the development process for the HIPPP application started with a rudimentary implementation for the purpose of testing the functionality of the logical control flow throughout the application. Subsequently, we embedded the accessibility principles, making the interface inclusive to users of any ability. An appealing design and user experience are in fact critical for the targeted user group. Therefore, we extended the DIME GUI models to include user affordances and other user experience (UX) principles.

Figure 1 shows one of the many iterations of the application's landing page. We see in Figure 4 the structure of its top-level GUI model: It contains a nested GUI model for the LandingPageHeader, an action SIB for the Search functionality, and a display Box for the static “About us” information. The LandingPageHeader is itself a GUI model shown in Figure 4.

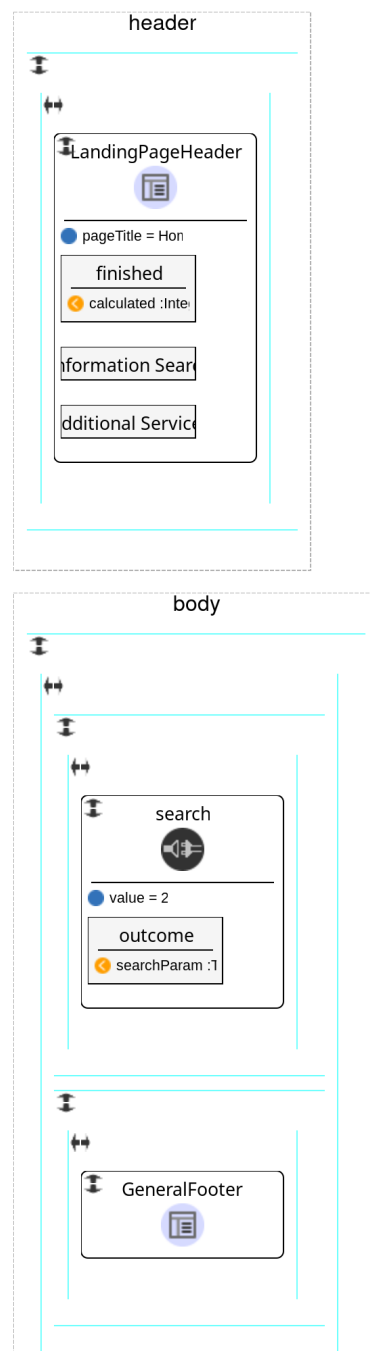


Figure 4. Top-level GUI model of the HIPPP landing page (Figure 1) in DIME.

3.1.2. Process Modelling

Process models in DIME describe the business logic of the application under design. This model type captures the control flow and the data flow of the application in a hierarchical fashion. This way, processes that implement finer granular tasks can be reused inside other processes, eliminating duplication of native code development and ensuring uniformity within one application and across applications.

Developing process models using DIME as a modelling environment provides a number of benefits to a developer. Its approach to no-code/low-code development and inbuilt model validation reduces the development time and improves error handling. If an application requires a process/component that is not yet available in DIME, it is possible to develop it or include it from an external provider and expose it to the DIME users as a service-independent building block (SIB). The action SIB for the Search functionality

shown in Figure 4 is such a SIB. For the HIPPP application, most of the processes have been designed using native DIME functionality, and only a small number of atomic SIBs have been developed in a low-code fashion. For example, as the front-end application communicates with the evaluation pipeline through a HTTP request to an API endpoint, a small palette of atomic SIBs for HTTP handling was created. As they are generic, they are now available to every developer that needs that kind of communication.

The orchestration of the evaluation pipeline is also defined by workflows designed as DIME process models. For each different stage of the pipeline, a request to a different API endpoint would be needed. Without abstraction and parameterisation, each such call would require a specific atomic SIB. Instead, we abstracted the structure of these calls and designed a parameterised model that includes all the necessary information for the SIBs so that it is now reusable in both this and other applications.

3.1.3. Data Modelling

The main purpose of the data model in a DIME application is the same as class diagrams in the unified modelling language (UML): to define accurate representations of simple and complex data types and to define their relations and dependencies. Differently from the UML practice, where a class diagram is “just” a depiction and the implementation of the type, classes and association need to be conducted manually in some programming language, the DIME data model has all the information to generate automatically that code, and it is connected by the DIME platform with the SIBs, the processes and the GUI models. This is why we call it an integrated modelling environment: a lot of the manual steps that in most modelling tools require hand-coding, management of dependencies and checks or testing are here automatically supported by the DIME platform, which implements a rigorous “write things once” approach, combined with automated checks and significant support for reference management, integration, reuse and code generation.

In the HIPPP application, the main complex data types are the user’s request acquired from the GUI, and the corresponding responses provided by the evaluation pipeline. The website URL that the user wishes to have evaluated is easily represented as an object that is sent to the evaluation pipeline for analysis. For the evaluation, a process model orchestrates the stages, sending the appropriate object to that processing SIB and receiving a response in JSON format. The content and size of this response are different at each pipeline stage. So, we have again the case of a general pattern that is different case by case. Also here, this lends itself to abstraction: we define a generalised data model to be used by each stage of the pipeline, with the differences captured by parameterisation.

In this project, the data are stored externally to the application. The data for the evaluation pipeline are accessed externally, and the HIPPP user data concerns the user account information, including the expert users and web admin login.

3.2. AI—Model Driven Development

When creating the AI pipeline to evaluate the information contained in the websites, the first step is to specify the requirements for the evaluation system. With the DIME *blueprinting* feature, designers can define process models with a set of inputs and outputs without having to implement the actual functionality. This is perfect for sketching the logic of the overall application and of the individual processes, it is sufficient to discuss all the aspects of the design, e.g., creating the data model and the UI prototypes, and it can be then concretised and refined at need towards the final, fully implemented processes. For example, Figure 5 shows the blueprint of the HIPPP natural language processing (NLP) pre-processing pipeline, which is part of the pre-processing stage of the HIPPP evaluation pipeline of Figure 3.

When following the model-driven approach, using blueprint SIBs enables the developer to rapidly prototype the system from both a control flow and data flow perspective before implementing any SIB in code. Here, we see that the control flow is linear (the solid arrows), and we see which data artefacts are produced at each stage and where they are

used. Data flow analysis is therefore very simple and visual, while it is a complex task if performed on code. Having this blueprint view also makes it easy to identify multiple occurrences of atomic SIBs: they only need to be implemented in code once (low-code development) and can be reused at many stages of the overall pipeline by including the corresponding SIBs in the process models (no-code reuse).

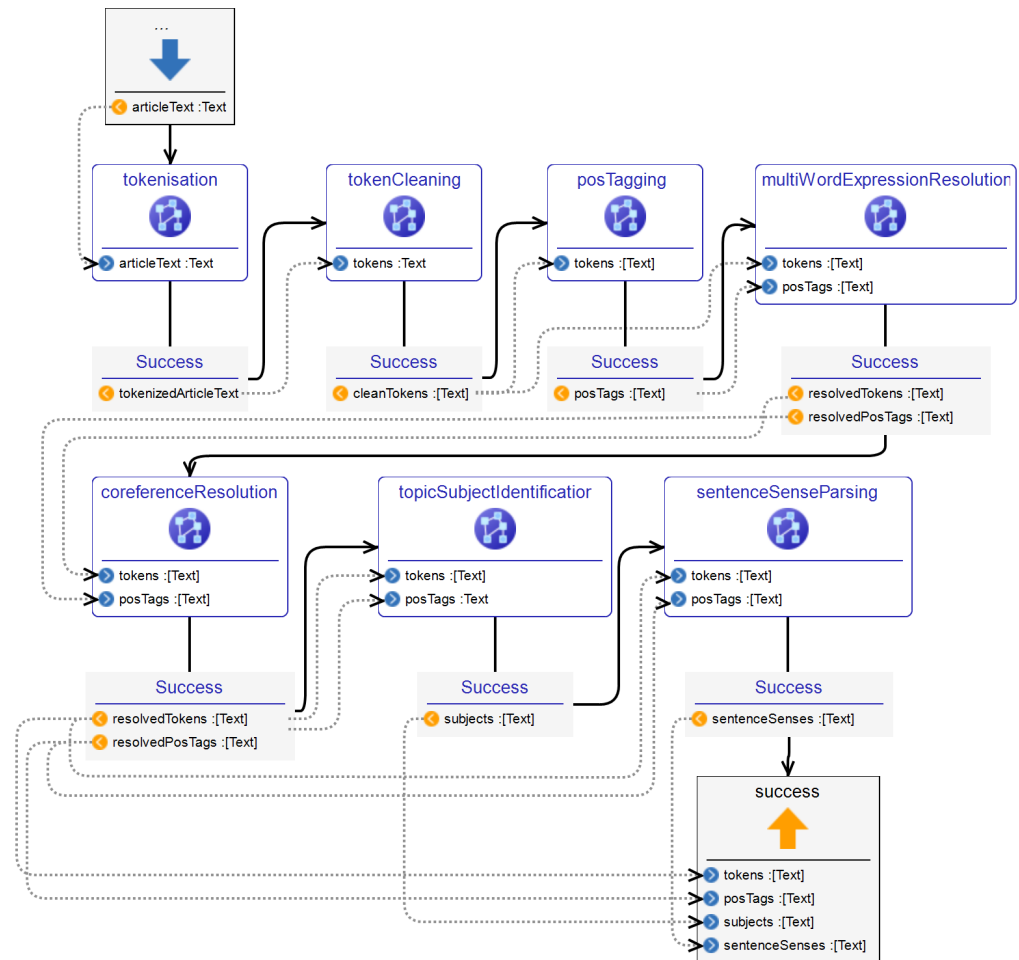


Figure 5. Blueprint of the HIPPP natural language processing (NLP) pre-processing pipeline in DIME.

At design time, the overall system functionality of the HIPPP AI evaluation pipeline was decomposed into its main sub-tasks (see Figure 6): website retrieval, web content extraction, data pre-processing, feature extraction, feature embedding, classification and incremental learning. Each sub-task was itself specified through blueprint processes, as shown in Figure 5 for the task of NLP pre-processing of the texts. The sub-models and SIBs contained in each such sub-task model belong to one of two types: *process models* and *intelligent process models*. Any function that implements an AI algorithm is an *intelligent process model*, and all the others are usual process models. Figure 6 shows the classification of the process models along their process types. As can be seen in the diagram, the type of some functions is clear. E.g., Classification and Feature Extraction are clearly related to the intelligence in the AI evaluation and are implemented through AI/ML technologies, while Website Retrieval, Data Pre-processing and the support process for Incremental Learning are clearly ordinary processes, with no AI/ML inside, others are amenable to choice. For example, Website content extraction could be implemented using either process or intelligent process models. For example, Feature Transformation/Embedding could be implemented either using statistical operators to summarise a set of data (process

model) or by training a recurrent neural network to create an embedding of that same set of data (intelligent process model). The decision is up to the design team, and the MDD approach supported by DIME could even allow both implementations, each a hierarchical process model encapsulated in a process SIBs, and let the users who compose the higher-level orchestration choose which one to include.

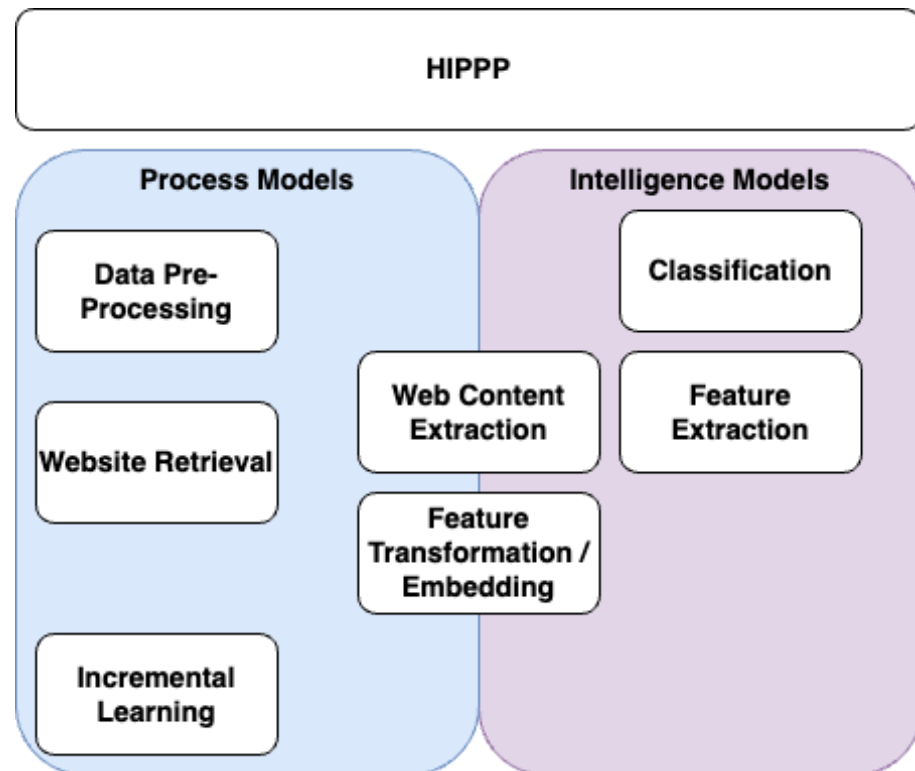


Figure 6. Process model types of the sub-tasks in the HIPPP evaluation pipeline. This figure breaks down how we categorised the different sub-tasks. Tasks such as pre-processing are conducted solely with traditional algorithms that do not need to be trained on data to perform their function and are developed using the procedure described in Section 3.1.2. Tasks such as feature extraction, etc., are AI-based methods, which followed the development of intelligent process models procedure in Section 3.2.1. Some tasks such as web content extraction can be implemented either using traditional process models or intelligent process models.

3.2.1. Development of Process Models

The process models within the evaluation pipeline were developed following the Agile Software development life-cycle. Once the described design structure and orchestration were stable, the refined process models were designed, and the individual components were implemented in Python and then tested, leading to several cycles of improvement that iterated the design refinement, implementation improvement, validation and testing loop. Once a functionality was locally satisfactory, the code was then wrapped in an API, containerised and integrated into DIME using the parameterised SIB approach described in Section 3.1.2.

The specific processes we wish to highlight here concern the development of intelligent process models, the data collection and labelling methods and the feature extraction/engineering task.

Development of Intelligent Process Models

Our development of the intelligent process models followed the established process guideline of the Machine Learning Workflow reported in Figure 7. We followed this best practice reference approach in the development steps 1 (Model Requirements) to 8 (Model

Deployment). The most crucial steps are the data collection, the data labelling and the feature engineering.

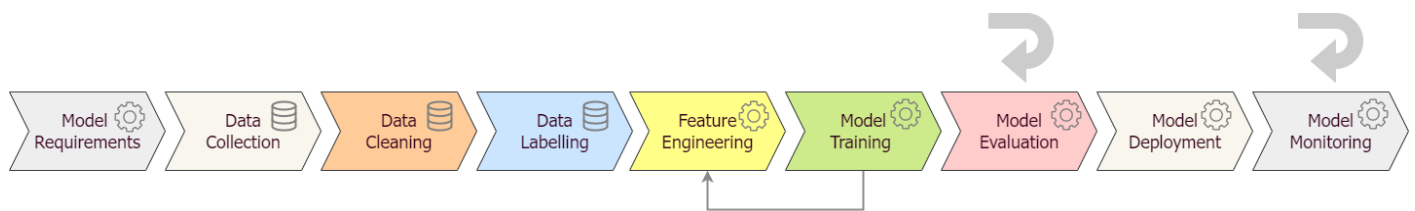


Figure 7. The reference Machine Learning Workflow as described by [22].

Data Collection and Labelling Methods

Several different sub-tasks are involved in solving the data collection and labelling problem. Although, in some cases, there were existing data sets which could be used, a number of data sets needed to be created from scratch. We ended up collecting URLs found through keyword searches on search engines, re-purposing existing resources such as free available datasets (<https://newsdata.io/datasets>, (accessed on 1 June 2023)) extracting links from social media such as extracting URLs from posts on the “Health” and “Cancer” subreddits and also scraping Wikipedia pages related to “Cancer”/“Health”. In order to obtain “gold standard” information sources, we leveraged domain experts within the UL Cancer Network (ULCaN) who curated lists of high-quality information sources. The collected data were then labelled/annotated for the various feature extraction tasks. This is an ongoing process, currently being carried out in collaboration with the ULCaN experts, to ensure the high quality of the data and of the annotation from both an AI and a medical perspective.

Feature Extraction/Engineering

In general, the set of features that are extracted from data to be input to the classifier can be designed manual or automated fashion. We chose to adopt a manual approach, taking the QUEST criteria as a point of reference. Figure 8 shows an extract from the QUEST system for scoring WBHI with respect to authorship. The other criteria are also described and scored in a similar, discrete-point fashion.

Figure 9 shows how we automated this scoring system using three AI methods to extract relevant features from the data: web content extraction (WCE), named-entity recognition (NER) and finally relation extraction (RE) to extract features, which automate the task in conjunction with a rule-based classifier.

Works such as AUTODISCERN [15] follow the automatic feature engineering approach, whereby the researchers score the several WBHI pieces using the DISCERN tool (an alternative to QUEST), then input the text from the WBHI and their respective scores into an ML model, and the model automatically learns the set of features to extract in order to perform the classification. While this is a valid approach, the features are typically not human-readable/interpretable, and they may not generalise well for the problem we have. Undertaking feature engineering manually, as we did, involves significant additional work, but it greatly aids in explaining how the system reaches its results. We feel that in the context of WBHI, where the result of this process influences health decisions by the users, high-quality evaluation is extremely important. The engineered features are still extracted using AI algorithms, but they are models trained to perform narrow AI tasks that are human-interpretable. For example, as seen in Figure 9, named-entity recognition takes a sequence of tokens (words) and classifies them as being part of a named entity (i.e., a person’s name or a job title) or not.

Authorship	(Score x 1)
0 – No indication of authorship or username	
1 – All other indications of authorship	
2 – Author’s name and qualification clearly stated	

Figure 8. Excerpt from the QUEST tool: scoring WBHI with respect to authorship.

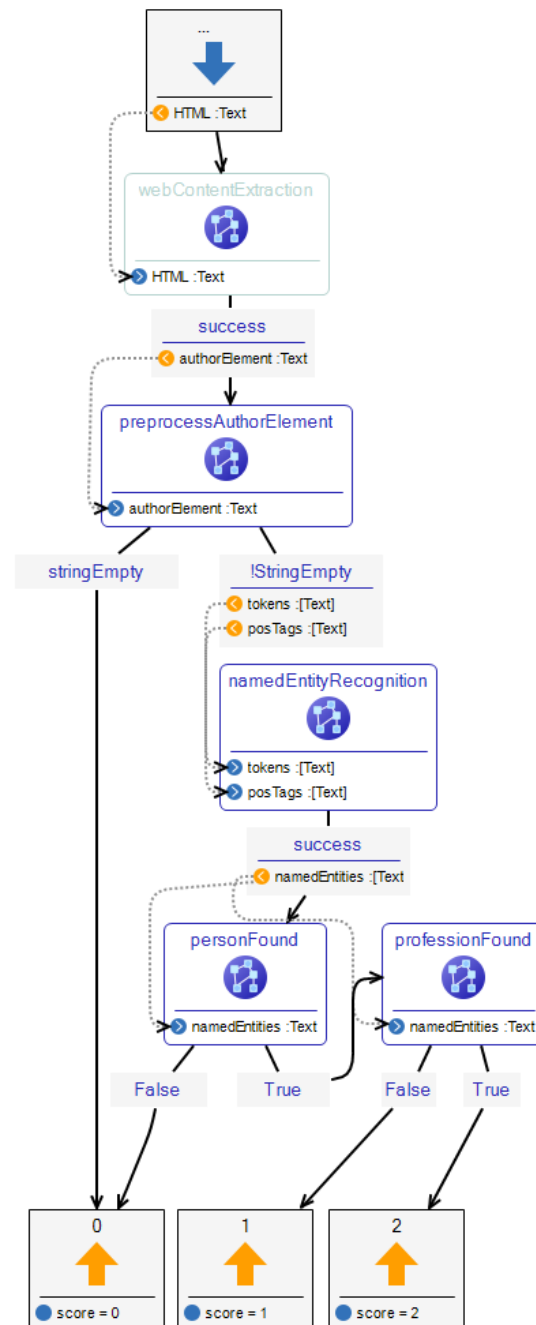


Figure 9. Overview of the feature extraction process for QUEST authorship classification modelled in DIME. Solid black lines denote control flow, dotted lines denote data flow.

3.2.2. Incremental Learning

As mentioned in Section 2.2.6, when the evaluation of a WBHI resource does not meet the confidence threshold, it is referred to experts for evaluation/labelling. In practice, this means that the results of all the ML classifiers in the pipeline are stored in an object store

along with the raw HTML, and the reference to it is added to a queue. This queue feeds the expert labelling portal front-end application. The queuing system sends a single example to multiple experts via the expert portal. When the experts have labelled it, the resulting labels are also stored. Resorting to several experts for each WBHI resource is part of the quality annotation policy: when a single example has been labelled by multiple experts, the inter-annotator agreement score is calculated. If the agreement is sufficiently high, along a threshold that is parameterised and thus easily changeable, the newly labelled data are passed to the incremental learning environment and used to re-train the existing models. This way, we implement a continuous learning and improvement process. A high-level overview of this process can be seen in the “after deployment” component of Figure 10.

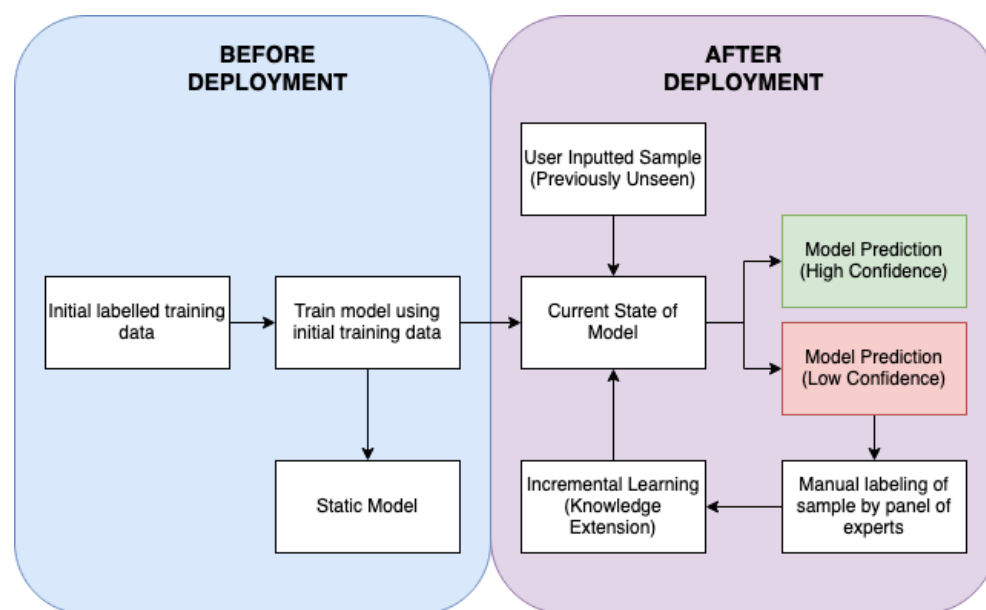


Figure 10. Incremental learning process for machine learning models in HIPPP.

The part of the HIPPP front-end application that implements the expert labelling portal will enable the domain experts to view the WBHI resources that cannot be labelled by the current classification model with sufficient confidence and to manually label them through an intuitive labelling system. A key requirement for the labelling system is to be efficient in terms of time and effort required by the domain experts. The panel of experts are donating their time, so a streamlined labelling process is key to the acceptance and the performance effort. We have created several prototype designs for this portal and taken user feedback into account in our iterative development.

3.2.3. Orchestration

As for the HIPPP front-end application, the AI evaluation pipeline also orchestrates a number of sub-processes designed as DIME process models and implemented through native or newly created SIBs. These SIBs manage an API call to the appropriate implementation, which is provided as a service running on a containerised instance of Python. We have taken a generalised approach to the development of the communication SIB seen in Figure 11. This atomic SIB takes the detail required to send a request to the evaluation pipeline services.

This approach to the orchestration of external services enables exact control of the process flow in DIME while still promoting generalisation.

3.2.4. Implementation of Pipeline Components

The individual pipeline components are implemented in Python and leverage a suite of libraries available in that ecosystem: trafilatura [23], tensorflow [24], scikit-learn [25] and

nlTK [26], among others. Docker is used for deployment. This is standard, and prior work describes in detail how to do so [27].

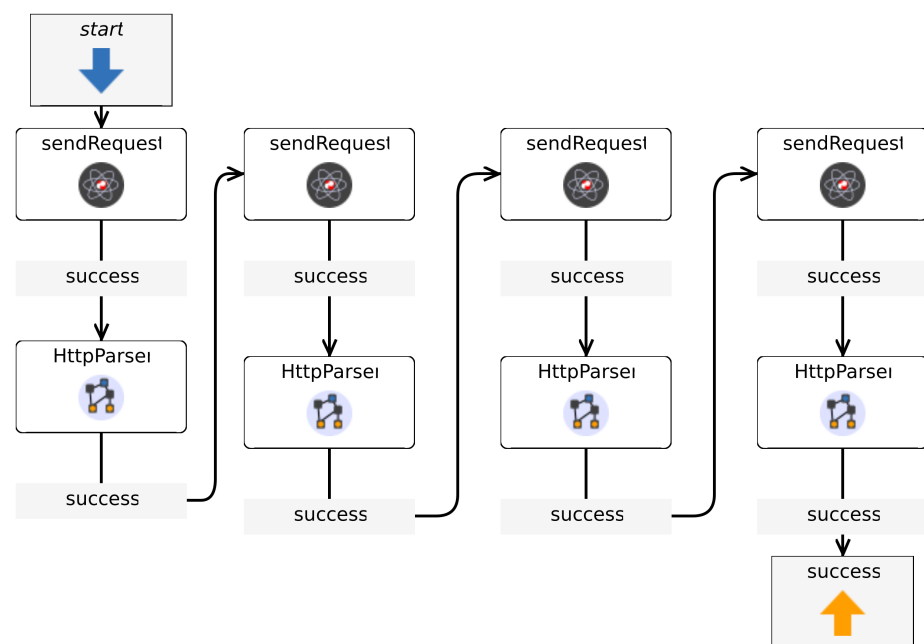


Figure 11. The top-level AI evaluation pipeline orchestration—DIME process model.

4. Results and Validation Approach

The results of the development are the HIPPP front-end application, of which we showed some screenshots and models, and the AI evaluation pipeline, of which we showed the design and organisation and several models. For this project, the main concern is the usability of the HIPPP system and the adequacy for the main user groups: the general public and the panel of experts.

As we are committed to agile development with continuous improvement, we will have successive rounds of updates and improvements based on feedback. It is clear to the ULCaN team that such applications are never fully finished and always in need of extension, e.g., when new needs arise, evolution through continuous improvement and maintenance when underlying platforms, like Docker, Python or the web application technology stack move to new releases.

Accordingly, we validate the HIPPP application on both main fronts: the web-based application and the end-users, and the evaluation pipeline.

4.1. Web Application and User Interface

A core requirement for this application is to have an appealing and accessible front-end UI. We aim to validate this by taking a person-centred approach [28], through the use and feedback about the UI by patient and public involvement (PPI) groups. PPI participants are recruited through the ULCaN network, including patients, carers, supporters and healthcare professionals. We have had so far several feedback rounds by the project members that are not system developers, and we are now ready to proceed to external validation. The initial PPI session will focus on the main features of the HIPPP application and the information evaluation pipeline.

The first PPI feedback session will be aimed at the calibration of the UI with respect to its design and accessibility. As already practised in the prior feedback rounds, detailed field notes will be taken during and after the feedback sessions. These will be analysed to identify potential changes to the user interface, and appropriate modifications will be agreed to and implemented. Future PPI group meetings will include clinical experts to provide feedback for the expert users portal and the labelling system, with the aim to gain

insights on the current design of that part of the HIPPP portal and on the use of the labelling system. Through a series of PPI sessions, we aim to achieve a validated UI implementation that is appealing and accessible to both target user groups.

4.2. Evaluation Pipeline

The WBHI evaluation pipeline is currently functioning, but not yet optimised, for the specific kind of data sets we are facing. Right now, we are not greatly concerned about the performance, as we are growing the set of labelled data sources, and we are building the initial classification model. Adopting the QUEST standard evaluation criteria has significantly simplified our design decisions so far, and it also ensures compatibility in principle and comparability with the other WBHI data sets manually classified along QUEST.

We expect the emergence of additional desirable criteria over time, either for performance improvement or for precision and specialisation, and we will in that case extend the current set of algorithms, SIBs and processes accordingly.

In terms of validation, so far we have evaluated the correctness/plausibility of the AI algorithms individually on the current data sets. For the task of web content extraction, we have built and labelled a data set of 1500 samples that have been labelled and annotated, achieving positive initial results on test sets. With respect to tone detection, we have collected and labelled a similar number of samples and are currently evaluating a variety of models. However, we wish to collect and annotate more data to ensure the models and the evaluation of those models are more robust.

Going forward, the validation/evaluation activity will proceed along two lines of action:

- *Empirical evaluation of the pipeline output* following ML model evaluation best practices. Here, each ML component will be evaluated using a testing regime, which will assess the models with respect to classification accuracy, robustness, interpretability of output and reproducibility of results. All models will be tested on labelled data sets previously unseen by the models (i.e., not used during model training or validation). A range of statistical tests will also be carried out along with in-variance tests, directional expectation tests and minimum functionality tests.
- *Validation of training data collection processes through PPI groups.* Once we conclude the first phase of data collection and base model building, the training data collection process will be validated through PPI groups. The aim is to get feedback on the kind of WBHI channels through which people access information in order to ensure that there is good coverage for these types of channels in the training data. Subsequent PPI group meetings will help validate how the classifications are displayed and ensure maximum comprehensibility of the evaluation for the target audience.

An example of the question is whether users feel that receiving the compact classification (e.g., reliable/not reliable) is sufficient, or whether they prefer also an explanation of how the evaluation pipeline reached its conclusions. Here, we see a clear trade-off between the simplicity of use and the level of detail of the answer.

4.2.1. Time Efficiency Study

As a key aim of the HIPPP project is to create a system that efficiently evaluates WBHI, we conducted a study to evaluate and compare the evaluation time of the system versus that of a human expert evaluator.

Experimental Setup

In order to gather data for the time taken for a human expert to evaluate a sample of WBHI with respect to the QUEST framework, we first recruited a senior internist to conduct the evaluations. We subsequently trained them in how to apply the QUEST framework to evaluate WBHI. In order to conduct the study, we then provided them with a tool for recording their evaluation scores and timings along with five samples of WBHI (unknown to them) from five Irish news organisations/blogs which contained articles pertaining to the diagnosis, treatment or prevention of colon cancer. The timing began as soon as the

evaluator launched the sample of WBHI in their web browser and concluded when they had recorded scores for each of the seven QUEST criteria to give an accurate recording of the actual time spent evaluating a given sample of WBHI.

For comparison, we ran the same five WBHI samples provided to the human evaluator through the HIPPP evaluation pipeline using two configurations. The first configuration evaluated the seven QUEST criteria sequentially, one after the other (akin to the procedure a human would apply the QUEST framework). The second configuration was optimised for evaluation time, whereby it evaluates the seven QUEST criteria in parallel, the trade-off being a larger processing/memory footprint. The experiment for evaluating the HIPPP pipeline was conducted on a machine with an Apple M1 Pro 8-core (CPU) and 16 GB DDR5 RAM, without GPU acceleration. In order to mitigate the effect that any performance issues on the machine may have on the results, each configuration of the HIPPP evaluation pipeline was run on the five samples 10 times, with the results being averaged.

Experimental Findings

The time efficiency study yielded the following results:

Across the five WBHI samples, the human evaluator took a mean time of 229 ± 54.81 s, the sequential configuration of the HIPPP pipeline took a mean time 16.83 ± 4.8 s and the run-time optimised HIPPP configuration took a mean time 2.76 ± 0.76 s, as seen in Figure 12. This equates to a time efficiency improvement of 92.65% (i.e., a factor 13.6) and 98.79% (i.e., a factor 83) for HIPPP and the run-time optimised HIPPP configuration versus that of an expert human evaluator.

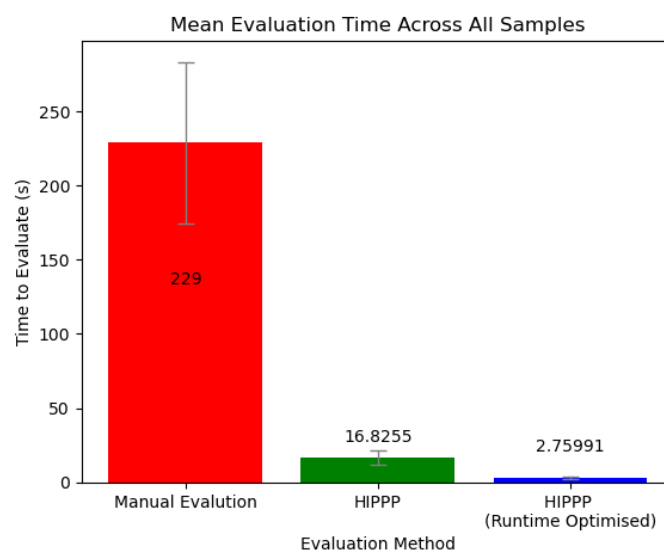


Figure 12. The mean evaluation time for each of the three evaluation methods (Manual, HIPPP, HIPPP Runtime Optimised), across the five samples of WBHI.

As a part of this study, we also investigated the effect that the length of an article has on the evaluation time both for a human evaluator and the HIPPP pipeline. The result of this can be seen in Figure 13. We fitted a linear regression line to each of the evaluation methods. In each of the three cases, it appears that there is a linear relationship between the number of tokens and the time taken to evaluate the documents. Intuitively, it would make sense that longer documents take more time for all methods of evaluation; however, it appears that the effect is more significant for human evaluators. Given the steeper line in the context of manual evaluation (red line in Figure 13), it would appear that there is a larger coefficient in the function that defines the relationship between the number of tokens and the evaluation time. As a result, it is possible in the case of very long documents that the performance gap between manual evaluation and HIPPP would be even wider. However, given the small sample size, we are unable to make a definite assertion as to whether this is the case.

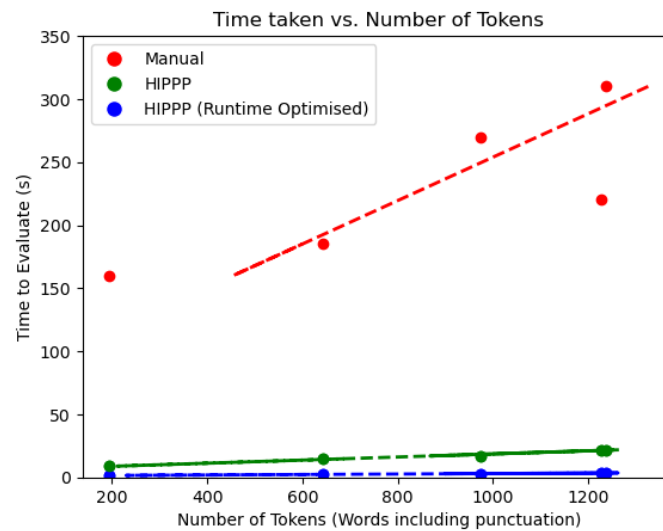


Figure 13. A plot illustrating the relationship that the number of tokens (words including punctuation) has with the evaluation time for each of the three evaluation methods.

An important caveat to the time efficiency study is that it assumes the real-time availability of a human expert evaluator at any time, for immediate evaluation. The reality, however, is that persons qualified to perform WBHI evaluation using the QUEST framework are typically unavailable to do so due to both clinical and other commitments. Therefore, in the context of real-world applications where a patient or a member of the public requires a piece of WBHI to be evaluated, the delay between receiving a result from an always online computational approach (such as HIPPP) and a human expert evaluator would be much larger, likely in terms of hours or days, i.e., no request would be evaluated until a qualified expert finds free time to devote to this task.

5. Discussion

5.1. HIPPP Web App and Reusability

The HIPPP web-based application has received positive feedback from domain experts throughout the development process. By leveraging the rapid development methodologies, we have been able to closely align the needs and requirements of our domain experts throughout the process.

In terms of GUI, for example, we started with a much less colourful skeleton GUI, where the bare interaction fields were visible, in order to validate the control flow and the application logic. As a result of the feedback session, we added background images, changed the colour scheme to the “friendly” perceived yellow with flowers and added a number of information fields like the “About us” one shown in Figure 1.

Also as a result of this iterative approach, we identified a number of “template” functionalities that led to the creation of a small number of parameterised reusable components consisting of both atomic process SIBs and GUI models. An example of this can be seen in Figure 11, where we reuse the same SendRequest SIBs to communicate with a number of different AWS endpoints. A more detailed view of this SIB is shown in Figure 14, where it becomes clear that the input/output signature is very abstract and can be concretised to many concrete usages.

In this sense, the promise of generalisation, simplification, increased reuse of the same code, and the ease of understanding the application logic (both control flow and data flow) and the data model by using the LC/NC DIME model-driven development platform has been validated in practice.

The emergence over the series of iterations of the reusable generalised templates has also led to a complete refactoring of the application, which now has less code, fewer bespoke SIBs and cleaner processes and is thus easier to understand, maintain and reuse.

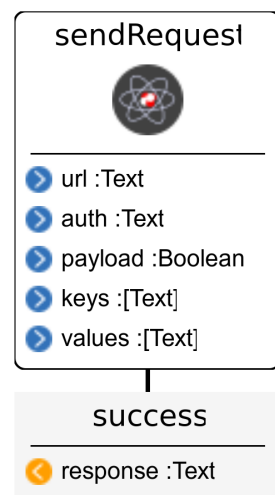


Figure 14. The SendRequest atomic SIB used to communicate with the HIPPI-AI evaluation pipeline.

5.2. Evaluation Pipeline

All the individual process models for the AI pipeline have been developed and validated, and they are currently being integrated into the DIME environment.

With respect to the intelligent process models, their respective iterative development life-cycle has led to significant improvements. This is best illustrated by the creation of models for web content extraction.

Rule-based methods such as *trafilatura* [23] were tested with initially satisfactory recall. However, we found the precision of such approaches to be lower than required as, for example, they also extracted ads and text snippets from other articles as being part of the current article body. To ensure the quality of the data extracted from web pages, we tested other techniques from the literature, like [29,30] among others; however, none of them met the requirements for our use case. Therefore, we are currently developing a novel graph neural network-based approach to solving the specific problem. It is yielding promising results on smaller data sets. In parallel, we are currently in the process of building a bespoke data set for WBHI content extraction and are progressing through the data labelling phase.

5.3. Contributions

In the context of this work, we assess our contributions as follows;

- Regarding domain-specific languages (DSLs) for health and medical research support, our contribution involves expanding the capabilities of the DIME platform. This expansion involved creating a GDSL and integrating a corresponding set of services for that GDSL. This enhancement to DIME's modelling capability will enable health and medical researchers to develop their own web applications or seamlessly integrate them into the design and development process of such applications more effectively.
- Concerning the data management cycle for integration and long-term reuse, the HIPPI user-facing application will function as a crowd-sourcing tool. It will collect user-provided web pages, storing them for future utilisation. The primary purpose is to enhance the application's performance through incremental learning with human input. Additionally, it can provide researchers with insights into the health information sources used by patients and the general public.
- In terms of technologies for the evolution and customisation of IT platforms for health, the HIPPI project incorporates a substantial co-design element. This approach enabled us to incorporate familiar concepts from the healthcare and medical domain into the system, tailoring the system to meet the specific requirements and expectations of stakeholders from the medical and health domain.

5.4. Next Steps

The public-facing application has received positive feedback from the domain experts, and we aim to continue the verification process of its design by resorting to external PPI groups.

The next milestone for the application is the implementation of the domain expert labelling portal. We are currently evaluating different approaches to GUI design, as there is a trade-off between adopting a simpler, more generic design, easier to implement with GUI models and a bespoke GUI, which may streamline the user interaction but requires a large number of rather complex custom GUI models. This kind of custom GUI modelling requires a number of textual descriptors that would be tightly coupled. This would not be a good model-driven development practice, so we are in the process of analysing how much “bespoke” is needed to support an efficient expert labelling portal without impairing the maintainability of this part of the application.

We received ethical approval to undertake PPI group sessions to validate our training data collection practices and are preparing to recruit participants and undertake those sessions. The questionnaires and interviews will provide data leading to potential changes to the HIPPP application to improve the user experience of the target users.

We will also continue to develop the intelligent process models and integrate them into the DIME environment and the HIPPP web application. Once the full-scale HIPPP web application will be made available to the general public, we will enter the monitoring and continuous learning phase, where we will evolve and further develop both the platform and the ML models.

Author Contributions: Conceptualisation, C.B., A.J.D., D.K. and T.M.; methodology, C.B., A.J.D. and D.K.; software, C.B. and A.J.D.; validation, C.B., A.J.D. and D.K.; data curation, C.B. and D.L.; writing—original draft preparation, C.B. and A.J.D.; writing—review and editing, T.M., D.L. and D.K.; visualisation, C.B. and A.J.D.; supervision, T.M.; project administration, T.M. and D.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was conducted with the financial support of the Science Foundation Ireland Centre for Research Training in Artificial Intelligence under Grant No. 18/CRT/6223 and University of Limerick Health Research Institute ULCan grants Pillar 1 and Pillar 4. For the purpose of Open Access, the authors have applied a CC BY public copyright license to any Author Accepted Manuscript version arising from this submission.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Battineni, G.; Baldoni, S.; Chintalapudi, N.; Sagaro, G.G.; Pallotta, G.; Nittari, G.; Amenta, F. Factors affecting the quality and reliability of online health information. *Digit. Health* **2020**, *6*, 2055207620948996. [CrossRef] [PubMed]
2. Frequency of Internet Usage-CSO-Central Statistics Office. Available online: <https://www.cso.ie/en/releasesandpublications/ep/p-isshh/informationandsocietystatistics-households2019/typeofinternetactivities/> (accessed on 1 June 2023).
3. Fong, T.; Wang-Gillam, A.; Waqar, M.A.; Jeffe, D.B.; Kehlenbrink, L.; Gao, F.; Govindan, R. A survey of Internet utilization among patients with cancer. *Support. Care Cancer* **2011**, *19*, 1183–1190. [CrossRef]
4. Taioli, E.; Joseph, G.R.; Robertson, L.; Eckstein, S.; Ragin, C. Knowledge and prevention practices before breast cancer diagnosis in a cross-sectional study among survivors: Impact on patients’ involvement in the decision making process. *J. Cancer Educ.* **2014**, *29*, 44–49. [CrossRef] [PubMed]
5. Silberg, W.M.; Lundberg, G.D.; Musacchio, R.A. Assessing, Controlling, and Assuring the Quality of Medical Information on the Internet: Caveant Lector et Viewor—Let the Reader and Viewer Beware. *JAMA* **1997**, *277*, 1244–1245. [CrossRef] [PubMed]
6. Charnock, D.; Shepperd, S.; Needham, G.; Gann, R. DISCERN: An instrument for judging the quality of written consumer health information on treatment choices. *J. Epidemiol. Community Health* **1999**, *53*, 105–111. [CrossRef] [PubMed]

7. Boyer, C.; Selby, M.; Scherrer, J.R.; Appel, R. The Health On the Net Code of Conduct for medical and health Websites. *Comput. Biol. Med.* **1998**, *28*, 603–610. [[CrossRef](#)] [[PubMed](#)]
8. Cuan-Baltazar, J.Y.; Muñoz-Perez, M.J.; Robledo-Vega, C.; Pérez-Zepeda, M.F.; Soto-Vega, E. Misinformation of COVID-19 on the internet: Infodemiology study. *JMIR Public Health Surveill.* **2020**, *6*, e18444. [[CrossRef](#)] [[PubMed](#)]
9. Robillard, J.M.; Jun, J.H.; Lai, J.A.; Feng, T.L. The QUEST for quality online health information: Validation of a short quantitative tool. *BMC Med. Inform. Decis. Mak.* **2018**, *18*, 87. [[CrossRef](#)] [[PubMed](#)]
10. van de Ven, G.M.; Tuytelaars, T.; Tolias, A.S. Three types of incremental learning. *Nat. Mach. Intell.* **2022**, *4*, 1185–1197. [[CrossRef](#)] [[PubMed](#)]
11. Margaria, T.; Steffen, B. Extreme model-driven development (xmdd) technologies as a hands-on approach to software development without coding. In *Encyclopedia of Education and Information Technologies*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 732–750.
12. Norman, D. *The Design of Everyday Things: Revised and Expanded Edition*; Basic Books: New York, NY, USA, 2013.
13. WAI. *Accessibility, Usability, and Inclusion*; WAI: Wakefield, MA, USA, 2010.
14. Prieto, V.M.; Alvarez, M.; Cacheda, F. Analysis and detection of Soft-404 pages. In Proceedings of the 2013 3rd International Conference on Innovative Computing Technology, INTECH 2013, London, UK, 29–31 August 2013; pp. 217–226. [[CrossRef](#)]
15. Kinkade, L.; Allam, A.; Krauthammer, M. Autodiscern: Rating the quality of online health information with hierarchical encoder attention-based neural networks. *BMC Med. Inform. Decis. Mak.* **2020**, *20*, 104. [[CrossRef](#)] [[PubMed](#)]
16. Joury, A.; Joraid, A.; Alqahtani, F.; Alghamdi, A.; Batwa, A.; Pines, J.M. The variation in quality and content of patient-focused health information on the Internet for otitis media. *Child Care Health Dev.* **2018**, *44*, 221–226. [[CrossRef](#)] [[PubMed](#)]
17. Arif, N.; Ghezzi, P. Quality of online information on breast cancer treatment options. *Breast* **2018**, *37*, 6–12. [[CrossRef](#)] [[PubMed](#)]
18. Torrey, L.; Shavlik, J. Transfer learning. In *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*; IGI Global: Hershey, PA, USA, 2010; pp. 242–264.
19. Boßelmann, S.; Neubauer, J.; Naujokat, S.; Steffen, B. Model-driven design of secure high assurance systems: An introduction to the open platform from the user perspective. In Proceedings of the 2016 International Conference on Security and Management (SAM 2016), Las Vegas, NV, USA, 25–28 July 2016; pp. 145–151.
20. Merkel, D. Docker: Lightweight linux containers for consistent development and deployment. *Linux J.* **2014**, *2014*, 2.
21. Minin, H.C.; Alemán, J.J.; Sacramento, C.; Trevisan, D.G. A WYSIWYG editor to support accessible web content production. In Proceedings of the International Conference on Universal Access in Human-Computer Interaction, Los Angeles, CA, USA, 2–7 August 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 221–230.
22. Amershi, S.; Begel, A.; Bird, C.; DeLine, R.; Gall, H.; Kamar, E.; Nagappan, N.; Nushi, B.; Zimmermann, T. Software Engineering for Machine Learning: A Case Study. In Proceedings of the 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice, ICSE-SEIP 2019, Montreal, QC, Canada, 25–31 May 2019; pp. 291–300. [[CrossRef](#)]
23. Barbares, A. Trafalura: A Web Scraping Library and Command-Line Tool for Text Discovery and Extraction. In Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations, Bangkok, Thailand, 1–6 August 2021; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 122–131.
24. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-scale machine learning on heterogeneous systems. *arXiv* **2016**, arXiv:1603.04467.
25. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
26. Bird, S. NLTK: The natural language toolkit. In Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions, Sydney, Australia, 17–18 July 2006.
27. Chaudhary, H.A.A.; Margaria, T. Integration of micro-services as components in modeling environments for low code development. *Труды Института системного программирования РАН* **2021**, *33*, 19–30. [[CrossRef](#)] [[PubMed](#)]
28. Yardley, L.; Morrison, L.; Bradbury, K.; Muller, I. The person-based approach to intervention development: Application to digital health-related behavior change interventions. *J. Med. Internet Res.* **2015**, *17*, e4055. [[CrossRef](#)] [[PubMed](#)]
29. Wang, Q.; Fang, Y.; Ravula, A.; Feng, F.; Quan, X.; Liu, D. Webformer: The web-page transformer for structure information extraction. In Proceedings of the ACM Web Conference, Lyon, France, 25–29 April 2022; pp. 3124–3133.
30. Zhou, Y.; Sheng, Y.; Vo, N.; Edmonds, N.; Tata, S. Simplified dom trees for transferable attribute extraction from the web. *arXiv* **2021**, arXiv:2101.02415.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.