Hybrid Human-Machine Information Systems for Data Classification

Inauguraldissertation

zur

Erlangung der Würde eines Doktors der Philosophie

vorgelegt der

Philosophisch-Naturwissenschaftlichen Fakultät

der Universität Basel

von

Shaban Shabani

Basel, 2023

Genehmigt von der Philosophisch-Naturwissenschaftlichen Fakultät
auf Antrag von

Prof. Dr. Maria Sokhn, Dissertationsleiterin
Prof. Dr. Heiko Schuldt, Fakultätsverantworlicher
Prof. Dr. Dominique Genoud, Korreferent

Basel, den 20.06.2023

Prof. Dr. Marcel Mayor, Dekan

to my family

Abstract

Over the last decade, we have seen an intense development of machine learning approaches for solving various tasks in diverse domains. Despite the remarkable advancements in this field, there are still task categories that machine learning models fall short of the required accuracy. This is the case with tasks that require human cognitive skills, such as sentiment analysis, emotional or contextual understanding. On the other hand, human-based computation approaches, such as crowdsourcing, are popular for solving such tasks. Crowdsourcing enables access to a vast number of groups with different expertise, and if managed properly, generates high-quality results. However, crowdsourcing as a standalone approach is not scalable due to the latency and cost it brings in.

Addressing the challenges and limitations that the human and machine-based approaches have distinctly requires bridging the two fields into a hybrid intelligence, seen as a promising approach to solve critical and complex real-world tasks. This thesis focuses on hybrid human-machine information systems, combining machine and human intelligence and leveraging their complementary strengths: the data processing efficiency of machine learning and the data quality generated by crowdsourcing.

In this thesis, we present hybrid human-machine models to address the challenges falling into three dimensions: accuracy, latency, and cost. Solving data classification tasks in different domains has different requirements concerning accuracy, latency, and cost criteria. Motivated by this fact, we introduce a master component that evaluates these criteria to find the suitable model as a trade-off solution. In hybrid human-machine information systems, incorporating human judgments is expected to improve the accuracy of the system. Therefore, to ensure this, we focus on the human intelligence component, integrating profile-aware crowdsourcing for task assignment and data quality control mechanisms in the hybrid pipelines.

The proposed conceptual hybrid human-machine models materialize in conducted experiments. Motivated by challenging scenarios and using real-world datasets, we implement the hybrid models in three experiments. Evaluations show that the implemented hybrid human-machine architectures for data classification tasks lead to better results as compared to each of the two approaches individually, improving the overall accuracy at an acceptable cost and latency.

Acknowledgements

First and foremost, I am deeply grateful to my supervisors, Prof. Dr. Maria Sokhn and Prof. Dr. Heiko Schuldt for their continuous support, precious feedback, and guidance during my PhD studies. I would like to also thank Prof. Dr. Dominique Genoud for reviewing this dissertation as an external expert.

I would like to thank all my colleagues with whom I had the opportunity to work and collaborate on various projects and publications related to this thesis. A special thank you goes to my colleagues from the City-Stories project for the insightful collaboration: Lukas Beck, Laura Rettig, and Loris Sauter.

I would like to express my gratitude to the members of the Databases and Information Systems (DBIS) research group at the University of Basel. I extend my thanks to Ralph Gasser, Ivan Giangreco, Silvan Heller, David Lengweiler, Ashery Mbilinyi, Mahnaz Parian-Scherb, Lukas Probst, Luca Rossetto, Dina Sayed, Philipp Seidenschwarz, Florian Spiess, Alexander Stiemer, and Marco Vogt, for the thoughtful discussions, constructive feedback, and the pleasant moments spent together at various DBIS group events.

Over the past few years of this journey, I worked with colleagues at HES-SO Valais-Wallis and HE-Arc Neuchâtel. I would like to thank Fabian Cretton, Nicole Glassey, Maximiliano Jeanneret, Zhan Liu, Alex Olivieri, and Camille Pellaton, for the great time and fruitful collaboration in our joint projects.

Finally, I would like to thank my family for their unconditional support and for cheering me up during my educational journey.

This work was partly supported by Hasler Foundation in the context of the City-Stories project, under Grant number 17055, which is also thankfully acknowledged.

Contents

List of Figures

List of Tables

List of Examples

Part I

Introduction

1

Introduction

Traditionally in computing, humans have been considered as service consumers. Advances in modern computing systems have enabled humans to become service providers as well, where computational problems are partially or completely solved by outsourcing them to humans, bringing human-based computation into action [QB11]. Hence, the boundary between humans and machine learning systems has become more blurry, and this is observed in collective intelligence, social computing, and computer-supported cooperative work. In these hybrid systems, humans and automated systems work together to address problems that are difficult to solve individually [DDG+17; DCL+21]. The hybrid approach aims to build more efficient and effective systems. Here, we consider crowdsourcing as the service provider in data-driven computing scenarios. Keeping the human input in the loop helps in leveraging both scalability in terms of latency and cost, and improves accuracy by controlling and maintaining the quality of work from the crowds.

1.1 Motivation

Machine learning based systems have been used and comprehensively applied to solve various problems in computing. However, even for the most advanced machine learning models, some tasks are complex to be solved and accuracy is the issue. In contrast to this, these tasks are trivial for humans [GMJM+16]. This has brought the importance of crowdsourcing as a field, where problems are simply outsourced to a group of people. In terms of granularity, such problems are grouped as *macrotasks* where multiple workers are asked to work on the same assignment (e.g., translating a book [HRB14], transcribing a video [Fur16],

writing software [MCH+17]), or *microtasks* which are small and short-time consuming tasks (e.g., tagging an image, labeling a sentence). Due to the rise of online crowdsourcing platforms, the focus has been more on micro-tasks crowdsourcing, where crowd workers get paid for solving small tasks called Human Intelligence Tasks (HITs). This has seen application in different data processing problems such as categorization: e.g., mapping products to an existing collection of products [SRY+14], labeling: e.g., image or video tagging [VAMM+08; AD04a], data cleansing: e.g., deduplication (finding multiple objects that refer to the same real-world thing [CMI+15]), analysis: e.g., sentiment analysis [LLO+12], etc.

While the accuracy of solving tasks through crowdsourcing can be higher compared to computer algorithms for aforementioned tasks, there are two major scalability issues that arise: *latency* and *cost*. Humans are way slower than computers, and outsourcing all tasks in large datasets becomes expensive. As an example, an entity resolution problem for maintaining a database of hotels with 800K records becomes costly and very slow. For illustration, applying purely crowdsourcing for 800K HITs and assuming the cost per HIT is $0.05, it would cost at least $40K, and it would take days to solve the problem. Considering machine learning-based and human-based systems for solving problems, the issues fall in three dimensions: *accuracy*, *latency*, and *cost*. The challenge is finding the trade-off. As a solution, we consider hybrid human-machine information systems. Once trained, machine learning models are fast and have low costs for data processing but face the issue of accuracy, we leverage the humans' collective intelligence as a feedback mechanism in the loop of data computing scenarios.

This thesis proposes hybrid human-machine learning designs [DDG+17] as optimal solutions to tackle the accuracy problem at lower cost and latency. Through the application of a profile and location-aware crowdsourcing [ABI+13] in the cases where machine learning algorithms fail to perform with high accuracy. This feature will increase the performance of hybrid intelligence data problem-solving approaches, by assigning tasks to the people that have relevant knowledge as a feedback mechanism in the loop of data computing scenarios.

1.2 Problem Statement

The advancement of digital technologies has led to the rapid growth of data online, creating challenges for effective and efficient data processing. In recent years, the field of machine learning and artificial intelligence has seen tremen-

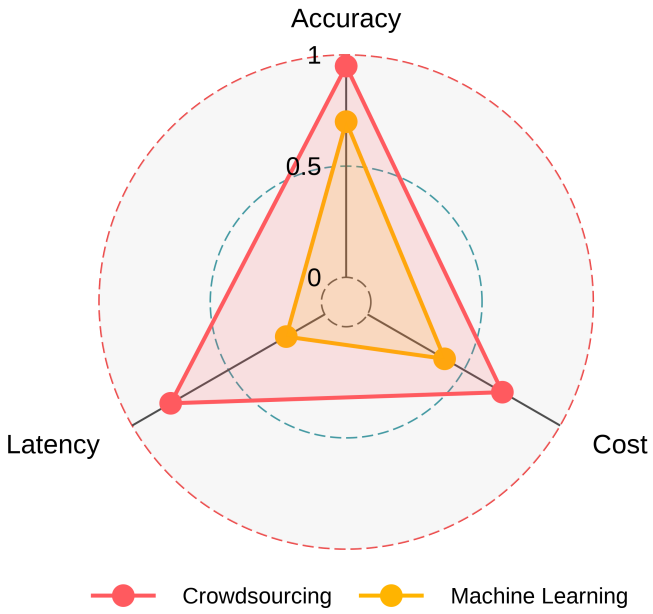


Figure 1.1 Crowdsourcing vs Machine learning: Fundamental challenges when solving data-related tasks.

dous applications useful in handling large volumes of data in low latency, however, fully algorithmic data processing approaches fail short to perform with acceptable accuracy. On the other hand, if properly managed, humans have the potential to process data with high accuracy, however, that is not scalable in terms of cost and latency. In what follows, we will describe the three-dimensional challenges of effective large data processing tasks, depicted in Figure 1.1.

1.2.1 Accuracy of Automated Data Processing Approaches

For many production environments, a significant shortcoming of the machine learning models is the accuracy and reliability, especially when these models are not trained for the type of content at hand. There are tasks that are inherently hard for computers to perform on their own, such as tasks related to text processing like sentiment and emotion analysis, and text summarization; image and video processing like transcription, context, or emotional understanding; data processing like data cleaning and filtering, entity resolution, etc.

In the text mining field, analyzing the context, opinion, and emotion in the text is an issue that is very important for enterprises. For instance, for businesses that operate on the Internet, it is precious to understand what their clients think of the services they provide. Sentiment analysis [Liu12] is the field of study that aims to analyze people's opinions, sentiments, and emotions behind a written

text. It uses natural language processing (NLP) techniques and computational linguistics and has a wide range of applications outside of computer science. With the growth of social media, the importance of sentiment analysis has raised as well, gaining the attention of research communities on advancing the development of algorithms. A major focus has been on social media analysis, on detecting the polarity of written text, for instance, classifying reviews and social network posts (e.g., Twitter) as positive, neutral, or negative. Many studies have approached the machine learning-based approach to address this issue [DZ14; SSS17; AAM+17], however, this remains challenging when it comes to the accuracy of the text mining algorithms. This is due to the complexity of the written text, online users have different language nuances and cues, and they make use of abbreviations, slang, and sarcasm. Indeed, detecting sarcasm is one of the most challenging tasks in the NLP domain [TSM14]. Another topic showing that automated text analysis is difficult is the detection of false information on the web [SSW+17; FO17]. One reason is that it is difficult to capture the writing style of false content. While some algorithms for automatic detection are based on writing style, on the same note, false news providers have taken counter measurements in changing their writing strategies to bypass the detection.

Advances in deep learning research have seen wide application in image classification and tagging, object detection, and sentiment segmentation [DK16; ZFW+17]. Deep neural network models trained on large datasets with thousands of classes can classify images with high accuracy. In reality, the needs are oriented towards custom classification scenarios in different domains. Creating a new labeled dataset is time-consuming and very costly, so training a model from scratch is not feasible. Therefore, adapting already trained models to specific classification scenarios is possible via pre-trained models and transfer learning [WKW16]. Pre-trained models are trained in a large and general classification context, and the knowledge is transferred to solving a new related problem. In situations when the application domain and the classes are very different from those that pre-trained models have been trained, domain adaption with retraining is necessary. This entails creating a dataset, labeling it with the new classes, and retraining the models with new data to even more tailor to the specific applications. In principle, the size of this dataset would be smaller than in cases when a model from scratch is to be developed. More frequently, classification tasks involve binary classification where the dataset record is assigned to one out of two classes, or multi-class problems where the data is assigned to one of the multiple available classes. However, there are situations where a

dataset entry can belong to one or multiple classes at the same time, and this is known as multi-label classification. Such tasks are more difficult for fully automated solutions, especially when more strict performance evaluation metrics are considered.

1.2.2 The Latency of Human Intelligence

While Artificial Intelligence (AI) has advanced and become part of human life, human capabilities still continue to exceed the AI-based solutions to various data analysis problems. Relating to the text analysis tasks explained earlier such as sentiment and emotion analysis, a human can easily understand the polarity and emotions behind a written text, if the text expresses a positive, negative, or neutral sentiment. Humans ultimately outperform NLP algorithms in understanding more complex tasks, such as detecting sarcasm in a text. The reason that makes sarcasm detection [TSM14] a very challenging task for NLP systems, is the difficulty of finding out the meaning of a sentence by only inspecting the words in a text. Humans are sarcastic when they convey a different message to what it literally means. While normal sentences contain at most one polarity, sarcastic messages can contain two sentiments of different polarity, both positive and negative. We find such a case in the sentence presented in Example 1.1.

Example 1.1 A sarcastic example

"My PC is so fast that my best time to have my cup of coffee is when the computer is booting."

Based on the words of this sentence, a positive polarity is determined, since the person is acknowledging that the computer is very fast. In reality, the person is being sarcastic at the computer booting process taking time as long as drinking a coffee. There are additional factors that contribute to such tasks being complex to analyze for computer algorithms, such as abbreviations, unknown words, slang, misspelled words, etc. Using computer-based linguistics approaches to understanding text is a challenging task in combating digital misinformation and digital disinformation topics. Considering the ubiquitous nature of false information and how that evolves and spreads via various channels, verifying algorithmically the content of the news is difficult as it additionally requires fact-checking. In recent years, fact-checking services have become the norm for journalists and are now also easily accessible by the public [BF17]. Moreover,

there have been innovative fact-checking initiatives [PEOOAP21; BMS+18] that provide resource guides to help the general audience navigate digital information content and train them to become fact-checkers themselves.

Image processing algorithms trained with good and sufficient data can solve various tasks with high precision. However, having good and enough data for training machine learning models is not always the case. Moreover, tasks that involve semantics and deeper understanding going beyond the visual features, require additional information and humans can easily solve these tasks compared to algorithms.

Considering the above presented challenges for AI-based approaches, human intelligence is the answer to having high-quality data. The main drawback is that humans are way slower than computers at processing data. Relying only on human feedback is not scalable in terms of time. However, research advancement in the human computation field has shown a high impact on developing several crowdsourcing platforms. Commercial platforms like Amazon Mechanical Turk[1] (AMT), Appen[2], MicroWorkers[3] have become popular as they have high availability of online paid crowd workers from different continents, providing a 24/7 service to solving various tasks. Moreover, such platforms offer efficient generation and deployment of crowdsourcing jobs via their application programming interfaces (APIs). For a job requester, it is possible to programmatically configure and deploy crowdsourcing micro-tasks, monitor the progress, and manage the crowd answers. Analysis of the activity logs from MTurk platform has shown that the number of micro-tasks, rewards, and requesters has been increasing over time [DFI18]. These factors contribute to lowering the latency in the process of retrieving and solving tasks via online crowd workers.

1.2.3 The Cost of the Crowd Wisdom

In addition to tasks that are trivial to be solved by a non-trained person, some tasks require specific knowledge in order to assess the problem with high quality. For instance, a requester might have a large collection of images of paintings from a particular period of art history. This task would not be suitable for the generally available crowd workers, but for a target group who might have skills related to the topics of the task. Modern crowdsourcing platforms offer a wide range of selection criteria spanning from general ones such as education, gender,

[1] https://www.mturk.com
[2] https://appen.com/solutions/crowd-management/
[3] https://www.microworkers.com/

age, and location, to more specific groups that are defined based on the interests and performance of the crowd workers, like image annotation, data extraction, and validation, sentiment analysis, etc.

There are additional options to make the process more effective. Task design mechanism [FKT+13; HWQ21] plays an important role in obtaining high-quality crowd contributions. Another key factor is incentive mechanisms. Having specific tasks, paid crowdsourcing scenarios would require higher costs in order to attract participation. In addition, task redundancy is another crucial factor for data quality control in crowdsourcing tasks. It allows assigning the same task to multiple participants and their contributions are aggregated to derive a final answer. While task redundancy is important for data quality, it has a direct impact on the cost aspect. Nevertheless, compared to experts, the cost of available non-experts on crowdsourcing platforms is way lower, hence lowering the burden of high labor costs. Studies [SOJ+08] have shown that asking multiple non-experts and aggregating their answers can be as good as relying on experts.

In parallel to commercial crowdsourcing platforms that use money as an incentive, there has been great interest in non-paid volunteering platforms which have other dimensions of motivating users to participate. In fact, numerous crowdsourcing projects rely on volunteers who mostly find the social good as the main incentive for participation. Most popular are the citizen science projects [WC11] where members of the public gather together to address real-world research problems. GalaxyZoo [RBG+09] is a large project where volunteers contribute to identifying galaxies in astronomical images. Another incentive for participation is entertainment. Game with a purpose (GWAP) [AD08] are popular crowdsourcing projects where participants contribute by answering tasks while playing the game. Additionally, learning [vAh13] can be an incentive for participation. Overall, the main advantage of volunteer crowdsourcing is saving costs, however, it is challenging to maintain the attraction for contributors.

1.3 Scenarios

In the following, we provide three scenarios that aim to illustrate how hybrid human-machine information systems can be deployed in real life. These scenarios describe the motivation and benefits of combining the human intelligence with algorithms that are capable of handling large volumes of data.

Combating digital misinformation through human-in-the-loop approach

Example 1.2 Scenario 1

Anna is a social media consumer, each day she spends a considerable amount of time on Facebook. As a proactive Facebook user, she gets the information that interests her and shares some useful articles and posts on her feed. At the beginning of COVID-19, Anna as many of us started to worry about how serious the threat of the virus really is. As everything around the virus was yet unknown, she wanted to be informed about what was happening. At the same time, the newsfeed on her Facebook was overloaded with different articles that made her read more. These articles were mainly related to the: origin of the virus, outbreaks of infections, dissemination of the virus, protection against it, and other similar topics. One day, she came across an article that had a catchy title: "Spraying alcohol or chlorine all over your body kills the coronavirus!". Anna thinks that this article is very useful and as a person who tries to raise awareness, she thinks that it would be useful to pass on this information by sharing it on her feed and even more share it on WhatsApp with her close circle.

This scenario describes how Anna uses social media channels to get informed about content that she likes doing for her daily activities. It illustrates a typical example of digital misinformation during the COVID-19 pandemic situation. She comes across false information that spreads over different media channels, and as many others, Anna is at a point where she is not aware that her actions can contribute even more to this process. Having a solution that can detect on time with high accuracy and prevent the spread of such digital misinformation, is crucial. In this scenario, accuracy and latency are critical, whereas cost is not.

Detecting false news in the political domain

Example 1.3 Scenario 2

John is an adult US citizen who has an interest in following the news and getting informed about the situation in his country. He regularly follows the news media mainstream, and he is active in reading political news on social media, especially on social networks. There, he follows different pages with news content, as well as public figures and politicians. As the country is getting closer to elections, John realizes an eruption of news articles appearing on his news feed. He is curious to read more news articles, especially if the news have catchy titles and attractive content. While reading,

he starts inspecting the websites that are publishing these news articles. Most of these articles are published by web portals that John does not remember following before. This fact raises his curiosity about checking further these news providers. John realizes that the content of news articles published on social networks is not in alignment with other media channels that he follows on TV. For some articles, he is very skeptical that the content is true at all. For some others, he realizes that the content is trying to make fun of public figures, especially political candidates. His thoughts that the content is not trustworthy are confirmed after he starts exploring the websites to which he was forwarded to read catchy articles on social networks. John realizes that many other articles on these websites have similar content, therefore he is trying to figure out how he can contribute to raising awareness for his circle so that they should not blindly trust the content of such news providers on social networks.

In parallel to being a typical social media news consumer, in this scenario, John is interested in being more active in providing feedback on published content and becoming a contributor. He is curious to inspect further the content appearing on his social network news feed. This scenario deals more with false information that relates to political context where accuracy is a challenging dimension, whereas latency and cost are both important. Therefore, the trade-off lies between the three dimensions.

Categorizing cultural heritage data

Example 1.4 Scenario 3

Sophia has been a resident of Basel since 1974, and she grew up in the Basel Spalen quarter. Sophia is working for the state archives of Basel. She is passionate about cultural heritage and the legacy of the cultural assets. Sophia and her colleagues are aware of the importance and the crucial role that heritage institutions play in transferring historical information between generations and civilizations. Hence, the institution where she is working is undertaking initiatives that motivate the digitization of the already existing archives and motivate the collection of new cultural heritage assets via online platforms. To date, they have successfully digitized all the archive content that has been collected by different local partners. For instance, they have digitized 140 thousand images. Despite her excitement, the collected images are poorly annotated, and missing tags and categories. Categorizing and tagging manually these images would not be scalable, as that is a time-consuming and costly task. On the other hand, she knows

many cultural heritage enthusiasts who are willing to annotate this data in their free time. One of them is Michael, who in addition owns valuable data, such as photo albums, or audio and video archives. He is aware that sharing this data can be of great cultural value and of public interest. Sophia is searching for a solution that would enable Michael to easily share the content he owns and contribute to the annotation process of the already existing collected data, with the knowledge he has.

This scenario describes how Sophia's excitement about contributing to the digitization of cultural heritage data is turning into a struggle. Data without proper annotation may lose their relevance in search engines and consequently lose their historical value as they cannot be found. Compared to the previous two scenarios, categorization and tagging of cultural heritage data does not require emergent reaction to solve data problems, therefore, the latency criteria is not critical, whereas accuracy is important with minimal costs.

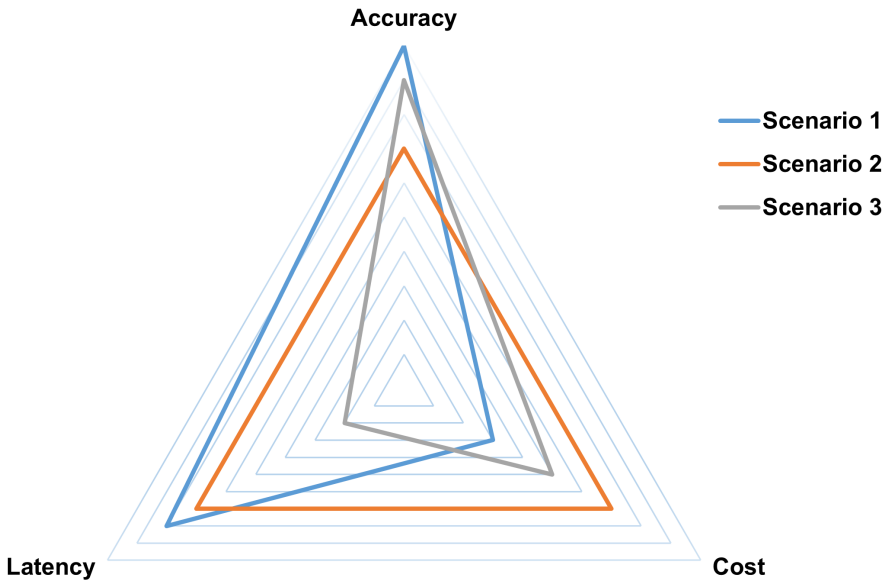


Figure 1.2 Trade-off dimensions of different scenarios: cost, latency, and accuracy.

For the above three described scenarios, Figure 1.2 illustrates the trade-off between the three-dimensional challenges: accuracy, latency, and cost. The closer the lines are to the edge, the more important the criteria is. For instance, for Scenario 1, accuracy has the highest priority followed by latency, whereas cost is less important. For Scenario 2, the three criteria (accuracy, latency, and cost) have the same level of importance, therefore, the trade-off is to optimize simi-

lar in three dimensions. For Scenario 3, obtaining accurate data at low cost is important, whereas time is less relevant.

1.4 Research Objective

Guided by the stated challenges that machine learning and crowdsourcing face individually, and motivated by the given scenarios, in the course of this research, our primary objective is focusing on:

> *How to design more effective hybrid human-machine information systems for solving complex tasks?*

With *complex tasks*, we refer to tasks related to data classification or data categorization that are difficult for machine learning models to address. If we link back to scenarios, the classification of online news articles into pre-defined classes (e.g., false or true) is considered a complex data classification task since it requires fact-checking skills and understanding the context of the problem. As another example, assigning categories to a historical image from a set of pre-defined categories (e.g., event, place, tradition) is considered a complex data categorization task because in specific cases it requires understanding the contextual meaning of the image.

We consider a hybrid system to be more *effective*, one that can outperform the individual systems (machine learning and crowdsourcing) with respect to the three conflicting criteria: *accuracy*, *cost*, and *latency*. This means that the hybrid solutions have to provide higher classification accuracy with a reasonable cost and latency.

We tackle these by utilizing and combining concepts from machine learning and crowdsourcing. Although hybrid human-machine approaches have been used before [DDG+17; DCL+21], these hybrid approaches have not specifically aimed at providing solutions that optimize the three conflicting criteria. Furthermore, we emphasize the importance of administering the *quality* of the data generated by crowdsourcing. While the challenge of data quality in crowdsourcing has been addressed widely [DKC+18], however, there is a gap in considering quality control in hybrid human-machine approaches. Integrating human feedback is designated to elevate the accuracy of hybrid approaches, therefore our aim is to implement and apply *profile aware crowdsourcing* methods to obtain accurate results from crowd participants.

Designing and implementing hybrid human-machine approaches requires understanding the data task. If we refer to our scenarios (see Section 1.3), we

observe that the three scenarios have different requirements with respect to the three criteria. There is no "one size fits all" hybrid design, therefore we propose three generic hybrid models that aim to cover scenarios to a considerable extent. Consequently, we set our next objective on:

> *How to select the most suitable hybrid human-machine design for a given specific scenario that has complex tasks to be solved?*

We address this by proposing a multi-criteria selection algorithm that assigns scores for the accuracy, latency, and cost criteria to each design. Additionally, weights that indicate the importance of the three criteria are used to optimize the selection process.

1.5 Contributions

Driven by the presented research objective, and motivated by the scenarios, in the following, we provide a brief outlook on the contributions made in this thesis:

- We propose three hybrid human-machine designs:

  1. *Human-in-the-loop designs* – The first method, named SAMS Human-in-the-loop (SAMS-HITL), complements features extracted with automated tools with features extracted based on the crowd input. This method leverages the fact-checking skills of humans to report important information about the dataset entries. This information is converted into features that are jointly integrated in the machine learning classification pipeline. The evaluations show that the SAMS-HITL method has a marked impact on the classification accuracy of the models [SCS+21]. This method is valuable for scenarios where the accuracy of data is of utmost importance at low latency, whereas the cost is negligible. The second method utilizes the cognitive skills of online crowd users to search for information on the web to fill in missing data. The information provided by the humans is injected into the feature generation extractor and integrated in the classification pipeline.

  2. *High-confidence switching hybrid design* – This method utilizes the performance of ensemble learning techniques to combine predictions from different algorithms and aggregation methods of classifications coming from the crowdsourcing component. The emphasis of this

design is on the decision-making model that decides whether a task needs human input or not based on the combined classification confidence of the machine learning models. This approach utilizes the low latency performance of machine learning models for tasks that can be solved with high confidence, whereas challenging tasks with low classification confidence are delegated for crowdsourcing. Evaluations of this design [SS18] have shown that finding the optimal parameters of the decision-making model leads to higher classification accuracy with acceptable cost. This design is suitable for problems where the three criteria accuracy, latency, and cost are equally important.

3. *Joint human-machine prediction design* – This method jointly combines the outputs coming from deep learning models and crowdsourcing to increase the accuracy of classification tasks. On the automatic classification part of this design, the focus is on building a multi-input model that uses visual features extracted from the images and textual features extracted from the metadata complemented with semantic features extracted from the text. On the human input part, the focus is on the aggregation of multiple crowd answers, which considers the estimated workers' reliability scores. Evaluations of this hybrid method [SSS20] show that it outperforms both deep learning and crowdsourcing when applied individually. This design finds application in scenarios where latency and cost of solving data problems is not an issue and the accuracy is not critical.

– We propose an entry component that evaluates the accuracy, cost, and latency criteria of the tasks to be solved and decides which of the three above proposed designs is suitable.

– We implement the above proposed architectures and set up the experimental environment for evaluation.

– We present the results of our quantitative evaluations, which show that combining machine learning classification models with human input leads to better performance compared to their performance alone. Moreover, this combination is shown as an optimal trade-off between accuracy, latency, and cost, and is applicable for various large-scale data classification tasks.

1.6 Thesis Outline

This thesis is divided into four parts, it consists of seven chapters and the bibliography. In Part 1, we describe the motivation and the vision behind the hybrid human-machine information systems. Part 2 provides an overview of the fundamentals upon which this thesis is written, introducing the individual components of hybrid workflows, i.e. supervised machine learning models and crowdsourcing. Part 3 contains the main contribution of this thesis, here we describe the concepts, implementation, and evaluation of the proposed architectures. In Part 4, we summarize the work and provide directions for future research in the context of hybrid human-machine information systems. Next, we provide a brief description of each chapter.

Part I – Introduction

Chapter 1 – In the first chapter, we introduce the challenges of solving data problems, both on machine learning based approaches and crowdsourcing approaches, individually. We then describe the motivation and vision behind hybrid methodologies, as a trade-off solution. Later, we state the problems, describe the scenarios, list the contributions of this research work, and describe the research methodology that was followed.

Part II – Background

Chapter 2 – This chapter describes the key concepts and technologies that this thesis is structured on. It will introduce the fundamentals of machine learning with a focus on supervised learning methods.

Chapter 3 – Provides an overview of the human-based computation methods, with a focus on the field of crowdsourcing. It details the benefits, issues, and challenges found in this problem-solving method.

Chapter 4 – Introduces the existing hybrid human-machine workflows, and emphasizes the benefits and potentials that these methods have to solve problems that are still complex to be solved solely by machine learning or crowdsourcing.

Part III – Hybrid human-machine information systems

Chapter 5 - Details the main contribution of this thesis. It describes the proposed hybrid human-machine designs:

(a) *Human-in-the-loop design* – The first method complements features extracted with automated tools with features extracted based on the crowd input. The human input is jointly integrated into the machine learning classification pipeline. The second method utilizes human input to fill in missing

information and generate features that are integrated into the classification pipeline.

(b) *High-confidence switching hybrid design* – This method utilizes the ensemble learning process by applying different aggregation techniques on generated predictions. To increase the accuracy of problem-solving, this method leverages human data classification skills whenever the machine learning models fail to perform with high confidence.

(c) *Joint human-machine prediction design* – This method jointly considers both the outputs coming from machine learning models and crowdsourcing to increase the accuracy of classification tasks, with a focus on data aggregation techniques. This design finds application in scenarios where latency and cost of solving problems are not an issue, but accuracy is critical.

Chapter 6 – Presents the experimental setup and the quantitative evaluations for each of the proposed hybrid designs (presented in Chapter 5). We describe the different use cases and the datasets used to apply the proposed methods.

Part IV – Conclusion and Future Perspectives

Chapter 7 – This concluding chapter provides a summary of the findings, contributions of this thesis, and provides directions for future research.

1.7 Publication Overview

Parts of this thesis have been published in peer-reviewed conferences and workshops. Table 1.1 gives the publication overview.

Table 1.1 Publication Overview

No.	Publication
1	Laura Rettig*, **Shaban Shabani***, Loris Sauter*, Maria Sokhn, Philippe Cudré-Mauroux, Heiko Schuldt. 2021. “City-Stories: Combining Entity Linking, Multimedia Retrieval, and Crowdsourcing to Make Historical Data Accessible." *In Proceedings of Web Engineering - 21st International Conference (ICWE 2021), 2021*. p. 521-524. *Best Demo Paper Award*.

2	**Shaban Shabani**, Zarina Charlesworth, Maria Sokhn, and Heiko Schuldt. 2021. “SAMS: Human-in-the-loop approach to combat the sharing of digital misinformation". *Proceedings of the AAAI 2021 Spring Symposium on Combining Machine Learning and Knowledge Engineering (AAAI-MAKE 2021)*, USA, p. 1-11.
3	**Shaban Shabani**, Maria Sokhn, and Heiko Schuldt. 2020. “Hybrid Human-Machine Classification System for Cultural Heritage Data". *In Proceedings of the 2nd Workshop on Structuring and Understanding of Multimedia heritAge Contents (SUMAC'20)*. ACM, New York, NY, USA, 49–56.
4	Zhan Liu, **Shaban Shabani**, Nicole Glassey Balet and Sokhn Maria. 2019. “Detection of Satiric News on Social Media: Analysis of the Phenomenon with a French Dataset". *In Proceedings of the 28th International Conference on Computer Communications and Networks (ICCCN 2019), Valencia, Spain, IEEE,2019, 2019 (ICCCN'19)*.
5	Alex Carmine Olivieri, **Shaban Shabani**, Maria Sokhn, Philippe Cudre-Mauroux. 2019. “Creating Task-Generic Features for Fake News Detection". *In Proceedings of the 53rd Annual Hawaii International Conference on System Sciences (HICSS'19), Maui, USA, 2019*.
6	**Shaban Shabani**, Maria Sokhn. 2018. “Hybrid Machine-Crowd Approach for Fake News Detection". *In Proceedings of the 4th IEEE International Conference on Collaboration and Internet Computing (CIC'18), Philadelphia, USA, 2018*.
7	Zhan Liu, **Shaban Shabani**, Nicole Glassey Balet, Maria Sokhn, Fabian Cretton. 2018. “How to motivate participation and improve quality of crowdsourcing when building accessibility maps". *In Proceedings of the 3rd International Workshop on Accessible Devices and Services, In conjunction with the 15th IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, USA, 2018*. *Workshop Best Paper Award*.

8	**Shaban Shabani**, Zhan Liu, Maria Sokhn. 2018. “Semantic Network Visualization of Cultural Heritage Data". *In Proceedings of the 1st International Workshop on Knowledge Graphs on Travel and Tourism, In conjunction with the 18th International Conference on Web Engineering (ICWE), Caceres, Spain, 2018.*
9	Alex Carmine Olivieri, **Shaban Shabani**, Maria Sokhn, Philippe Cudre-Mauroux. 2017. “Assessing Data Veracity through Domain Specific Knowledge Base Inspection". *In Proceedings of the 9th IEEE International Conference on Advanced Computer Science and Information Systems (ICACSIS), Indonesia, 2017.*
10	**Shaban Shabani**, Maria Sokhn, Laura Rettig, Philippe Cudre-Mauroux, Lukas Beck, Claudio Tanase, Heiko Schuldt. 2017. “City-Stories: A Multimedia Hybrid Content and Entity Retrieval System for Historical Data". *In Proceedings of the 4th International Workshop on Computational History, In conjunction with the 26th ACM International Conference on Information and Knowledge Management (CIKM), Singapore, 2017.*
11	Alex Carmine Olivieri, **Shaban Shabani**, Zhan Liu, Maria Sokhn. 2017. “Enhancing Cultural Heritage Information through Selective Crowdsourcing". *In Proceedings of the 9th IEEE International Conference on Advanced Computer Science and Information Systems (ICACSIS), Indonesia, 2017.*
12	**Shaban Shabani**, Maria Sokhn. 2017. “Gaming as a gateway: Ensuring quality control for crowdsourced data"". *In Proceedings of the 14th Conference on Cooperative Design, Visualization and Engineering (CDVE), Spain, 2017.*
13	Rima Kilany, Maria Sokhn, Hussein Hellani, **Shaban Shabani**. 2016. “Towards Flexible K-Anonymity". *In Proceedings of the 7th International Conference on Knowledge Engineering and Semantic Web (KESW), Czech Republic, 2016.*

Part II

Background

2

Supervised Machine Learning

This chapter provides an outline of the machine learning methods used in this thesis in order to solve data related tasks. The automated data-driven approach is an important component of the hybrid human-machine models we develop in this thesis. Since our focus is on the classification or categorization of data using labelled data, we describe supervised machine learning classification methods. Our application scenarios involve the analysis of text and images, therefore, we describe supervised machine learning models related to text and image classification. Additionally, multiple performance metrics are used to report the accuracy of classification algorithms, depending on the classification problem. In this chapter, we present the most common performance metrics reported in the literature.

2.1 Introduction

Machine learning in general is defined as computational methods using past information to make accurate predictions [MRT18]. The past information is usually digitized data that involves a human labelling task. The *quality* of human-labelled datasets is of utmost importance in order for the algorithms to make accurate predictions. Depending on the difficulty of the learning problem, the labelling task can be done by non-experts with little instruction, however, for certain tasks, expert knowledge is required to label datasets. Another factor that has an impact on the success of an algorithm's performance in the prediction is the *quantity* of the data. Prediction is a challenging task that becomes even more challenging when dealing with small-size labelled datasets. The limited size of training data can lead to unreliable and biased predictions done by classification

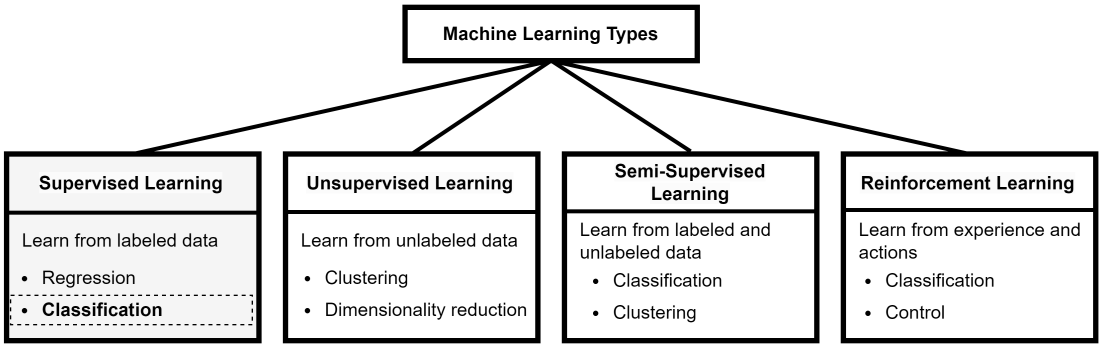


Figure 2.1 Overview of the major Machine Learning Types

algorithms.

An example of a learning problem is how to use a limited number of randomly selected images from a large dataset that contains historical images, where each image is labelled with a category, to accurately predict the category of unseen images. The examples (e.g., images) that the algorithm uses to learn, compose the training set, whereas each training example (e.g., image) is a training instance or sample. The difficulty of the classification task depends on the quality of the provided image labels, as well as the number of potential categories. For instance, in a dataset that has many categories to be learned for prediction, there is a need for a sufficient number of samples, and ideally equally represented. The *sample complexity* is an important notion in machine learning that is used for evaluating the number of training samples needed for the algorithms to learn a target function. In other words, sample complexity addresses the question: *Does the training set contain sufficient information so that the machine learning algorithm makes an accurate prediction?*

Machine learning focuses on designing efficient algorithms for extracting patterns from data to make accurate predictions [KT18]. These algorithms depend on the data being used, therefore machine learning techniques are data-driven methods that combine concepts from statistics, probability, and optimization [MRT18]. The selection and application of machine learning algorithms depend on the problem to be solved. According to the amount and type of supervision required during training, these algorithms can be classified into four major learning types: *supervised learning*, *unsupervised learning*, *semi-supervised learning*, and *reinforcement learning* [Sar21]. Figure 2.1 illustrates the major four learning types and their applications.

The most common and widely used in practice learning scenario is *supervised learning*, where the learner receives a finite set of training samples and predicts unseen samples. Given a set of input-output pairs $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, the objective

is to learn a function $f : X \rightarrow Y$ that maps inputs X to outputs Y. Here $\mathcal{D}$ is the *training set* with N samples, and x_i is the representation of the feature vector of the i-th example, while y_i is the corresponding *label* or *class*. For instance, the variable x_i may represent a historical image, while the label y_i could be the category of the image, e.g., place or person. Once the algorithm is trained, the ultimate goal is to predict the label y for any input x that is not present in the training set, which comprises the *testing set* [Bis13]. So, supervised learning aims at generalizing the observations from $\mathcal{D}$ to any new input, which is known as *generalization*.

Predicting the category of a historical image is a *classification* problem where the label y is discrete and comes from a finite set of k labels $Y = \{y_1, y_2, ...y_k\}$. The prediction problem could be also a continuous variable such as predicting the weather temperature and this problem is known as *regression*.

The second main learning scenario of machine learning is *unsupervised learning*, where only the inputs $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ are given without any corresponding target values. Here, the goal is to find interesting *patterns* in the data, sometimes called as *knowledge discovery*, or discover *groups* of similar examples within the data known as *clustering*. Another type of unsupervised learning is *density estimation*, where the problem is to determine the distribution of data within the input space. While in supervised learning, there are target labels and the performance of the learner can be quantitatively evaluated by comparing the predicted with the target output, in unsupervised learning, there are no target labels, therefore, it is difficult to quantitatively evaluate the performance of the learner.

Semi-supervised learning is the type of machine learning that stands between supervised and unsupervised learning. In this scenario, the learner is given both labelled and unlabelled data with the goal of predicting unseen samples. In contrast to supervised learning, here the amount of unlabelled data is large, and obtaining labels is expensive. The semi-supervised learners combine the available labelled data with unlabelled data during training, to capture the underlying data distribution and generalize to new data samples.

Another machine learning technique is *reinforcement learning*. It refers to the problem of finding optimal sequential decisions in order to maximize a reward [SB98]. In contrast to supervised learning, here the learner is not given examples of target outputs, but instead, discovers them by a process of trial and error. The learner takes actions and transitions between states. For an action y in a *state* x, the learner is provided with feedback and previous actions influence the future

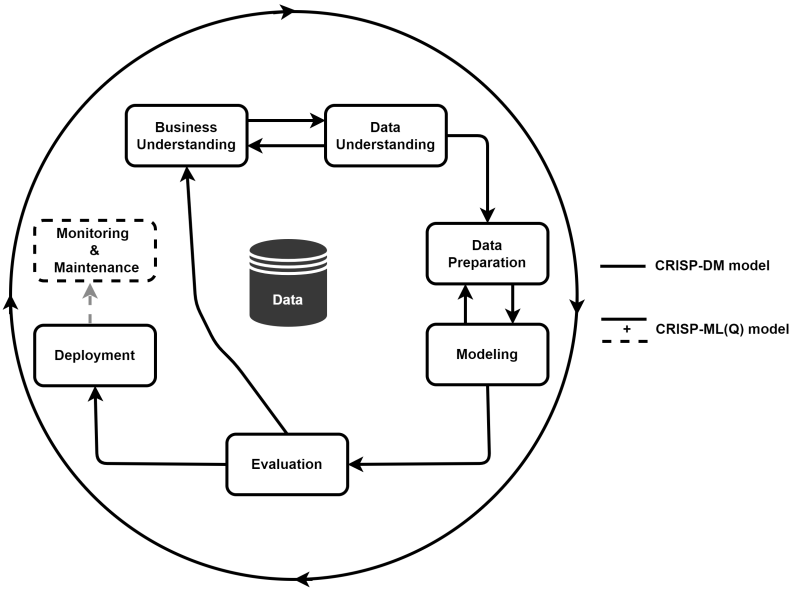


Figure 2.2 Illustration of the CRISP-DM [WH00] and its extension model CRISP-ML(Q) [SBD+21] which additionally has the "Monitoring & Maintenance" stage

actions. For each correct action, the learner is given a reward and punishment for a wrong one. For example, an agent is trained to operate in an environment where there are present obstacles. Correct actions are rewarded, whereas actions leading to collisions are penalized. Another reinforcement learning problem is playing a game [RN09] where the agent's goal is to achieve a high score by performing actions for which there is feedback for won and lost actions and that information gives reasonably accurate probability estimates of winning the next actions.

The above-mentioned machine learning problems can be further categorized into *offline* and *online learning*. In *offline* learning, the learner is trained by operating with a batch of training samples, whereas in *online* learning, it processes samples in a streaming mode. Naturally, reinforcement learning operates in an online mode, while supervised and unsupervised learning can operate both in an offline or online manner.

2.1.1 Stages of Learning Process

Here, we describe the most common top-level stages in the machine learning development lifecycle. The core stages follow the CRISP-DM methodology [WH00], a well-known industry-oriented guide for the development of data mining projects. This methodology defines a set of activities that are necessary for a product or service to be completed. These activities are organized in

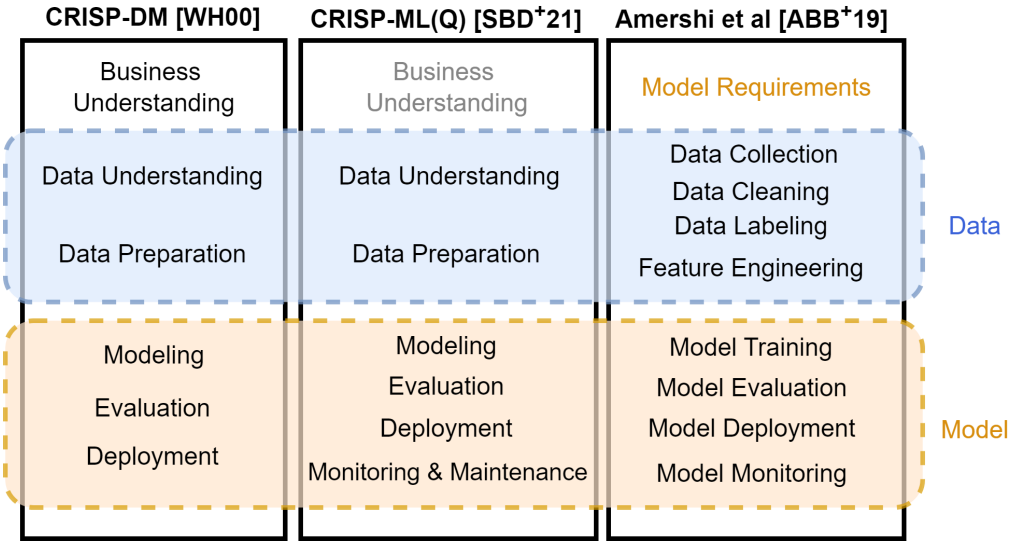


Figure 2.3 Overview of the three machine learning process models

six stages: *business understanding*, *data understanding*, *data preparation*, *modelling*, *evaluation*, and *deployment*. With the expanding applications of machine learning within organizations [LS20], efforts have been made to develop a machine learning process model. CRISP-ML(Q) [SBD+21] adapts and expands on the CRISP-DM model by appending another stage, the *monitoring and maintenance*. These two models are more industry-oriented, well defining the business requirements at the front and re-evaluating if necessary. Figure 2.2 illustrates a high-level overview of the two process models, their process stages, and their interactions.

Based on a case study conducted at Microsoft which observed software teams that develop AI-based applications, Amershi et al. [ABB+19] propose a nine-stage machine learning workflow process. This workflow has a more detailed set of phases when developing a machine learning project, and in practice, it is similar to the previous models (CRISP-DM and CRISP-ML(Q)). Figure 2.3 gives a big picture about the three process models. It can be observed that the three models have commonalities in between. All three models have stages that can be grouped into two main groups: *data oriented* and *model oriented* stages. Additionally, there are feedback loops between stages, e.g., between the model training and feature engineering stages.

Below, we focus and detail a subset of the stages within the *data* and *model* groups. Some stages in the model group such as *model monitoring & maintenance* are necessary for organizations that deploy ML projects in a production environment, therefore these are not in our focus. For each learning stage, we will illustrate the main concepts and their use in practice. These concepts and defini-

tions have been followed in our methods in Chapter 6. We found that the third model illustrated in Figure 2.3 is more suitable in practice for our conducted experiments.

2.1.1.1 Data Understanding

Assuming that the model and business requirements have been defined, the very first step is *data collection* and consolidation in a common schema. Data might come from different sources, in different formats, therefore transforming the data in one format is needed.

To understand better the classification problem for which a model will be implemented, it is crucial to inspect the raw data. Looking at the raw data can provide important insights that can drive the next stages of the learning process (data preparation, feature engineering, modelling, and evaluation), but also can lead to reviewing the preprocessing step. In cases when dealing with large datasets, it is not feasible to inspect the entire data records, but, it is possible to inspect a few records which are randomly sampled. For instance, when dealing with text data, reviewing this data will help understand what the columns are and their types. If some columns contain categorical values, it is better to convert them into categorical types. Some columns might contain numerical values, therefore understanding the distribution can help to decide which *data scaling* methods to apply.

Descriptive statistics give a great overview of the dataset and help to better understand the data. For columns that contain numerical values, statistical properties (mean, standard deviation, minimum and maximum values, and different percentile levels) can help understand their distributions. Furthermore, checking the distribution of dataset classes is important in order to understand how balanced the classes are. Highly imbalanced datasets, where some classes have more observations than others, need special handling. In such scenarios, during the evaluation of the model's performance, the interpretation of results would require consideration of the class distribution.

2.1.1.2 Data Preparation

Once the data has been collected and data is analysed and understood, the next step is data preparation. This stage involves *data cleaning*, which deals with incomplete (empty or missing part of the data records) or redundant data (duplicates). In such cases, the missing values are filled, or those samples are deleted from the dataset. In text analysis, it involves cleaning the text from special char-

acters and encoding it into standard format. Sometimes, when dealing with text data, columns of the dataset could be *decomposed* into more than one dedicated column. In some scenarios, *data enrichment* could contribute to the model's performance by augmenting existing data with data from external sources. In scenarios dealing with images, preprocessing would include transforming all images into one format and scaling into the same dimension.

2.1.1.3 Feature Engineering

Feature engineering refers to the process of transforming original raw input data into some new space of features that better represent the problem and will make it easier to solve. *Features* are a set of attributes or variables, usually represented as a vector, associated with the dataset *records*. For instance, if we have a task that will detect spam emails in our email server, we need to extract and develop features for understanding the text content of the email, as well as metadata such as the sender's details (name, email address), email server/provider, date, etc. A feature would be the domain of the sender's email server, which requires splitting the email address and extracting the domain part only. From the date information, the day of the week could be extracted, or the time/part of the day when the email was received. In cases when some features have a wide range of continuous values and especially when the number of dataset records is small, it might be required to create bins of ranges.

After *feature extraction* is done, the numerical values of different features might be on different scales, therefore, *data rescaling* is necessary. For instance, *feature normalization* also known as Min-Max scaling, is a technique that rescales the feature values in a range between 0 and 1, whereas *feature standardization* scales the feature values, so they have zero-mean and a standard deviation of one. Feature scaling has an impact on the algorithm's performance, both from accuracy and processing aspects. Sometimes, we can generate new *binary attributes* from existing features by converting them into 0 or 1 using a binary threshold. Additionally, categorical values can be encoded into numerical values, known as *categorical encoding*. Similarly, *label encoder* is used for normalizing the classes or labels.

Generated features have a major impact on the performance of classification models. Some features might be irrelevant to the problem, some others might be related to each other. Therefore, it is important to choose the most meaningful features, a process known as *feature selection*. Removing redundant and irrelevant features could help with improving the accuracy, reducing the train-

ing time, and reducing overfitting, as there are smaller chances for the model to decide on noisy data. Some algorithms such as Random Forests or XGBoost can provide the importance of each feature in the prediction task, therefore, evaluating the *feature importance* can additionally help in the feature selection process. In scenarios when dealing with numerous features, *dimensionality reduction* techniques such as Principal Component Analysis (PCA) [AW10] can compress the set of features. The goal of PCA is to reduce and extract the most important features, and that can potentially help to mitigate overfitting as well. *Overfitting* is a common issue in machine learning that occurs when a model is tuned to perfectly learn from the training data, causing it to memorize the training data instead of learning the patterns, therefore the model does not perform (generalize) well on unseen test data.

2.1.1.4 Modelling and Evaluation

The ultimate goal of developing the prediction pipeline is the acceptable accuracy of the algorithms on unseen data. In the modelling stage, the selection of the most appropriate models is done. The selection is driven by the given problem, the available data, and is sometimes influenced by the machine learning infrastructure. There are many algorithms available, and it is required to understand their characteristics, as there is no single algorithm that solves all problems. Additional important factors in the model selection stage are explainability, scalability, and complexity. It is recommended to start with simpler and low complex models and that would serve as a baseline for the evaluation [SBD+21]. Therefore, it is necessary to design a test environment to train and then evaluate a few algorithms on data for which we already know the labels.

Data splitting is an important part of performance evaluation. Depending on how much data is available, there are a few techniques for managing the data splitting. In cases when a large dataset is available, the simplest method is splitting into three different parts: *Training set*, *Validation set*, and *Testing set*. The size of the three parts can vary, for instance, an approach is to allocate 70% of the data for training, 15% for validation, and 15% for the testing set. This approach is simple and fast, however, it should be ensured that the three data sets have strong representation samples of the underlying problem, otherwise, high variance in the data can lead to different accuracy results.

The training set is solely used for training and fitting the algorithm's parameters. The validation set is used to tune the parameters of the learning algorithm, and during this process, the *hyperparameters* are optimized. Hyperparameters

are free parameters pre-configured and specified as input to the learning algorithm, e.g., the structure of the neural network. If optimally tuned, the classifier may be too specialized and report excellent performance on validation data, but poor results when applied on a testing set. Ideally, a classifier should be evaluated on examples that were not used to learn and fine-tune the model. Testing data is a set of examples used solely to assess the performance of a tuned classifier [KJ13]. Therefore, the testing set is usually locked in order to prevent peeking consequences and biased model effectiveness.

A simple validation approach has drawbacks because the validation set, which is a part of the data, remains fixed and is not used for training. This has an impact on the training process, especially for scenarios when the overall dataset is small and acquiring data is limited. A more advantageous and commonly used method of validation is the *k-fold cross-validation*. This method can help to estimate the performance of a learning algorithm with less variance than a single train-validate split. This method splits the dataset into k parts (e.g., $k=5$ or $k=10$), where each part is called a fold. The classifier is trained on $k-1$ folds and one fold is held out for evaluating the performance of the classifier. This process is repeated k times such that each fold of the dataset is given a chance for evaluation. This means, in each iteration an evaluation score is generated, and finally, we end up with k scores which can be summarized by calculating the mean and standard deviation of the performance scores.

The k-fold cross-validation method provides more reliable results compared to the simple train-test split approach and is recommended for smaller datasets where splitting the data into training, validation, and testing sets is not feasible. This method offers several advantages over the single train-test split because a sole test set has limited potential to characterize the variance in the data and uncertainty in the results; different test sets may produce significantly different outcomes. It ensures that the classifier does not become overly fitted to the specific variations of the training data, a process referred to as *generalization*.

2.1.2 Practical Applications of Machine Learning

Machine learning tackles a very broad set of practical real-world applications. We have mentioned the classification of historical images as an example application, however, there are a lot more scenarios [MRT18], some of which are described below:

– *Text or document classification* – this learning category includes problems

such as assigning a topic to a document, or categorizing text into pre-defined classes. A common classification task is spam filtering, where an email content analysis software scans the text of incoming emails to determine if it is routed to the inbox or filtered into the spam folder. Another well-known text classification is sentiment analysis, which aims to identify the polarity of the text content. Moreover, text classification can be applied to automatically flag improper content on the web.

– *Natural Language Processing (NLP)* is the intersection of machine learning and linguistics. It deals with processing and analysing large amounts of human language data to understand the text the same way human beings can. NLP has a broad range of applications such as part-of-speech (POS) tagging, named-entity recognition (NER), natural language generation, etc. POS-tagging is the process of categorizing each word in a text corpus as a particular part of speech (verb, noun, adjective, adverb, etc.) based on its context and use. NER is the process of detecting and labelling entities in a structured or unstructured text into pre-defined categories such as person names, addresses, locations, or organizations [SR09].

– *Speech processing* – this learning category includes different speech signal analysis such as speech recognition, text-to-speech, voice recognition, emotion recognition, etc. Speech recognition, also known as speech-to-text, enables the processing of human speech into a written format, whereas text-to-speech is the vice versa application that converts language text into speech. On the other hand, voice recognition is used to identify an individual user's voice, such as speaker verification and identification.

– *Computer vision applications* include a wide range of real-world applications such as image recognition, optical character recognition (OCR), medical imaging, biometrics, and content-based image retrieval [Sze10]. For instance, image recognition is one of the most common learning tasks that is used to identify objects, persons, and places in images. As an example, image recognition is prominently used in social media to automatically tag people in images, using face detection and recognition algorithms.

2.2 Text Classification

In the last recent years, the number of mobile and web-based applications has increased tremendously, generating a high volume of data on the web. Con-

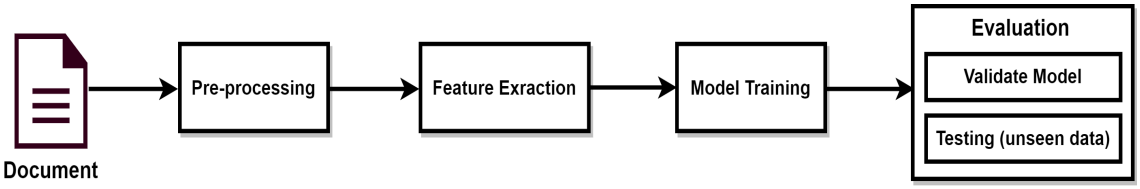


Figure 2.4 Overview of Text Classification Pipeline adapted from [KJMH+19]

sequently, the need and importance for efficient text mining applications have been raised as well. Classical text mining techniques have primarily focused on processing raw text data, while modern techniques have opened up opportunities for novel approaches to information extraction and linking with resources beyond the text, including multimedia data such as images and videos. [AZ12]. In most of the cases, text mining pipelines would follow the broader process of learning stages described in Section 2.1.1 Here, we provide a detailed and decomposed process specifically tailored for text classification scenarios, [KJMH+19], illustrated in Figure 2.4. It comprises four main blocks: text preprocessing, feature extraction, model training, and evaluation. There is additionally the dimensionality reduction, which is an optional block.

An input dataset consists of documents $\mathcal{D} = \{X_1, X_2, X_3, ..., X_n\}$, where $\mathcal{X}_i$ refers to a document in the dataset. Each document consists of sentences, and sentences consist of words. For each document, there is a corresponding class y_i from a finite set of k classes $\mathcal{Y} = \{y_1, y_2, y_3, ..., y_k\}$. The first step in the classification pipeline is text preprocessing. In this stage, methods such as data cleaning, removing stop words, and stemming, are applied to the raw text documents. The next stage is feature extraction. Machine learning algorithms expect that a text document is transformed and represented into a structured vector form, called the *feature vector*. The feature vector is an n-dimensional vector that contains features that describe and should be informative about the document. Feature extraction aims to improve the efficiency and processing of classification algorithms. Lastly, choosing the right model is important to finally evaluate its performance in the prediction task. In settings where the dataset results with numerous features, it is often preferred to further reduce its dimension. The key benefits of dimensionality reduction are i) *computational*: it compresses the initial feature set to speed up processing operations on the data; ii) *visualization*: high-dimensional input data are mapped into two- or three-dimensional spaces, such that the data can be visualized for Exploratory Data Analysis (EDA).

In the following, we describe the previously mentioned stages of the text classification pipeline.

2.2.1 Text Cleaning and Preprocessing

Text preprocessing is an important step that has a positive impact on the performance of a classification task. The aim is to remove noise and omit unnecessary content from raw text. Different methods can help to find more informative features.

2.2.1.1 Noise Removal

Text documents often include extraneous content, such as punctuation and special characters. Depending on the applied feature extraction techniques, punctuation and special characters can negatively impact the performance of the classification algorithms [UG14], and it is common in NLP applications to remove them. However, in certain scenarios, particularly in sentiment analysis, feature extraction requires the inclusion of plain text, including punctuation. This is the case with the application called Linguistic Inquiry and Word Count (LIWC) which is a tool that extracts various features related to the emotional, cognitive, and structural components in written speech samples [PBJ+15]. This tool also analyses the punctuation.

Example 2.1 An online user review example

"Great products for affordable prices. We will definitely buy again from this store!"

2.2.1.2 Tokenization and Segmentation

Tokenization is the process of splitting the text document into smaller units called *tokens*, which could be words, phrases, or symbols. Larger text can be separated into pieces larger than words, such as paragraphs and sentences, and this is referred to as *segmentation*. Tokenization is the process used to break down longer strings of text into individual words. For instance, for a comment posted in an online shop (shown in Example 2.1), using a *WordTokenizer*[1] generates the following tokens:

Example 2.1 (continued) Tokenization of the sentence

{'Great', 'products', 'for', 'affordable', 'prices', '.', 'We', 'will', 'definitely', 'buy', 'again', 'from', 'this', 'store', '!'}

[1] `https://www.nltk.org/api/nltk.tokenize.html`

If we are interested in removing punctuation, there are tokenizers such as *RegexpTokenizer*[2] that use regular expressions to handle this.

2.2.1.3 Stop Words

Text documents include many commonly used words in any language. In text mining applications, *stop word lists* are built to filter these words in the text preprocessing phase. Removing stop words is helpful since they are not specific or discriminatory to the different classes, hence the focus falls on other more important words. In Example 2.1, English stop-word lists would filter out the following stop words:

Example 2.1 (continued)	**Stop words list**
{'for', 'will', 'again', 'from', 'this'}	

2.2.1.4 Stemming and Lemmatization

Text documents contain words that for grammatical reasons use different forms, e.g., *write*, *writes*, and *writing*. The base form *write* is called the *lemma* for the word. *Lemmatization* is the process of grouping or transforming the inflected forms of a word in their base form (lemma) or dictionary form of the word. When searching text, the results often include documents containing variations of the searched keyword. For instance, when searching for the word "boat", we might get results for "boats", and "boating". The word "boat" here is the *stem* for all variations *boat*, *boats*, and *boating*. *Stemming* is the process of stripping the ends of the words (e.g., plural into singular). The aim of both stemming and lemmatization is to reduce the words to their base form [MRS08].

2.2.2 Feature Extraction

Text classification counts as one of the major application fields for machine learning algorithms. Text documents subject to analysis contain words and symbols of variable length. Raw text data cannot be fed into a learning algorithm because these algorithms expect numerical feature vectors of a fixed length as input. Therefore, feature extraction techniques extract numerical features from the content of the text documents.

[2] `https://www.nltk.org/_modules/nltk/tokenize/regexp.html`

A simple and commonly used technique is the *Bag-of-Words* (BoW) model that describes text documents by word frequency. It transforms text documents into a numerical feature matrix where each row represents a document, whereas each column is a word or token present in the corpus. This process is also known as *vectorization*. One of the limitations of the BoW model is that it does not consider the semantic relationship between words and sentences in the corpus.

Word Embedding as a feature extraction technique overcomes the shortcomings of the BoW models by considering the semantic aspect. Words with similar meanings have similar representations. Word Embedding algorithms consider each word in its context and generate a real-valued vector, encoding the meaning of each word, such that words that stand near each other in the vector space have similar meanings.

Another group of feature extraction techniques looks at the *sentiment* of the text data based on lexical approaches. Written text mainly contains facts and opinions, where facts can be verified, whereas opinions involve people's sentiments and feelings. These techniques aim at determining the polarity and subjectivity of a given text. Specific tasks require exploring the emotional, cognitive, and structural components of written speech. Features representing these aspects are important in scenarios that aim to assess the objectiveness of written text.

2.2.2.1 Term Frequency-Inverse Document Frequency (tf-idf)

Term Frequency-Inverse Document Frequency (tf-idf) is a BoW common technique in text mining that measures the relationship of words to documents. It is a method that combines Term-Frequency (tf) and Inverse Document Frequency (idf) to reflect the importance of a word in a document or corpus. tf-idf is used for feature extraction in text classification and is widely applied in information retrieval as a weighting metric in searching. For instance, many recommender systems in digital libraries are based on the tf-idf technique [BGL+16]. The Term-Frequency (tf) simply measures the frequency of a term, Equation 2.1, where $f_{t,d}$ is the number of occurrences of the term t in a document d divided by the total number of terms in that document.

$$tf(t,d) = \frac{f_{t,d}}{\sum\limits_{t' \in d} f_{t',d}} \tag{2.1}$$

On the other hand, Inverse Document Frequency (idf) measures how much information the term t provides, if the term is common or rare in a given document

corpus, Equation 2.2. It is calculated by dividing the total number of documents N by the number of documents in the corpus that contain the term t, i.e. document frequency df_t, and taking the logarithm.

$$idf(t, D) = log \frac{|D|}{df_t} \tag{2.2}$$

Thus, the *IDF* score of a rare term is high, whereas the *idf* score of a frequent term is expected to be low. The *tf* and *idf* can be multiplied to derive the new measure *tf-idf*, Equation 2.3. It measures how relevant a word is for a given document in a corpus of documents.

$$tf\text{-}idf(t, d, D) = tf(t, d) \times idf(t, D) \tag{2.3}$$

Depending on the classification task, the corpus of documents can become large, therefore extracting the most representative words as features requires a threshold. Typically, in the process of feature extraction, the top n words with the highest *tf-idf* weights are chosen. Alternatively, words with scores below or above a specified threshold may be filtered out [BYRN11].

2.2.2.2 Sentiment features

Many text classification tasks are related to understanding the sentiment of the text documents. Sentiment analysis is an important branch of NLP and text analysis that designs and implements techniques to determine if a given text has objective or subjective information, and determine the polarity, i.e., if the information is expressed in a positive, neutral, or negative way. The rise of social media popularity has led to massive subjective content shared on social networks (web forums, review sites, social media comment feeds, etc.), strengthening the application of sentiment analysis models for opinion mining. Today, fabricated reviews, false news, manipulated content, and malicious rumours spread at an extraordinary rate. Automated analysis based on linguistic approaches can help in assessing the data veracity. The analysis of sentiment expression has shown an important role, especially in the domain of false news [AVGR+21]. For instance, the approach of analysing the sentiment of shared content by online social network users has been demonstrated to be efficient in distinguishing real human accounts from social bot accounts [DKS14]. Furthermore, linguistic approaches explore the content of deceptive messages to associate language patterns and find the *predictive deception cues* [CRC15].

The *pattern.en* [DSD12] tool allows the extraction of sentiment features. It uses a lexicon of adjectives (e.g., good, bad, excellent, awful, magnificent, etc.)

Table 2.1 Polarity and Subjectivity scores from Example 2.2

Adjective	Polarity	Subjectivity
boring	-1.0	1.0
several	0.0	0.0
missing	-0.2	0.05
certain	0.22	0.57
sentence	-0.25	0.41

that occur frequently in online reviews, and these adjectives are annotated with scores for sentiment *polarity* (positive, neutral, negative), and *subjectivity* (objective, or subjective). For a given text, the *Sentiment* function analyses the adjectives it contains and returns the scores for the polarity and subjectivity. The *polarity* score ranges from -1.0 being negative to +1.0 being positive, whereas a score near 0 indicates neutrality. The *subjectivity* score ranges between 0.0 and 1.0, where a score towards 0 indicates high subjectivity whereas a higher value towards 1 indicates objectiveness.

Example 2.2 An online movie review example

"This movie was ranked at number one, and I have no idea why. For me, it was so boring that I had to watch it several times because I kept falling asleep and missing certain parts."

For instance, Example 2.1 is a positive review of an online web store. The *sentiment* function returns a score of 0.4 for polarity and a score of 0.625 for subjectivity. If we consider the Example 2.2 which is a negative review for a movie, the sentiment scores for polarity are -0.25, and +0.41 for subjectivity. The tool detects the adjectives used in the sentence, and for each adjective, there are pre-defined polarity and subjectivity scores. Table 2.1 describes the detected adjectives from the sentence and their corresponding polarity and subjectivity scores. The -0.25 polarity and +0.41 subjectivity scores are averages of the adjectives used in the sentence.

Additionally, grammatical *mood* captures the level of certainty expressed in a text. It refers to the use of auxiliary verbs (e.g., should, would, could, etc.) and adverbs (e.g., badly, extremely, absolutely, always, etc.) to measure uncertainty. The *mood* function analyses a text and returns one of the four categories: i) *Indicative*, if it presents facts or beliefs; ii) *Imperative*, if there are commands or warning messages; iii) *Conditional*, if conjectures are used; and iv) *Subjunctive*,

if wishes or opinions are expressed. The *modality* function evaluates the degree of certainty of a text with a value between -1.0 and +1.0, where values greater than 0.5 represent facts. For instance, a sentence *"I wish it was snowing!"* would be evaluated with a score of -0.25, a slightly modified sentence *"It is mostly likely going to snow."* would give a +0.31 score, whereas the sentence *"It is snowing."* has a score of +1.0.

2.2.2.3 Linguistic features

Language is a basic function to communicate, and words provide the core information in what message is delivered, identifying the emotions expressed, the degree of subjectivity, if a person is being honest or deceptive, social relationships, and personalities [TP10]. Linguistic Inquiry and Word Count (LIWC) [PBJ+15] is a computerized text analysis tool that counts words in a text concerning psychologically meaningful categories. It uses a dictionary of 6'400 English words and 90 defined categories:

- word count – the total number of words from the analysed text
- 4 *summary language variables* (analytical thinking, clout, authenticity, and emotional tone)
- 3 *general descriptor categories* (words per sentence, percentage of target words captured by the dictionary, and percentage of words longer than six letters)
- 21 *standard linguistic dimensions* (e.g., percentage of pronouns, prepositions, articles, negations, auxiliary verbs, etc.)
- 41 categories tapping *psychological constructs* (e.g., positive and negative emotions, anxiety, anger, sadness, social processes, etc.)
- 6 *personal concern* categories (e.g., work, home, leisure, etc.)
- 5 *informal language markers* (fillers, swear words, assents, netspeak, nonfluencies)
- 12 *punctuation* categories (periods, commas, colons, semicolons, exclamation marks, total punctuations, etc.)

When analysing a text, LIWC computes the word occurrences in each of the 90 categories, and one word can fall into several categories. The output is a feature vector of 90 scores. LIWC has been successfully applied in the deception

detection topic [MS09; OCC+11]. LIWC features have shown improvements in distinguishing suspicious from verified online posts on social media, by capturing well signals of persuasive and biased language in user tweets [VSJ+17].

2.3 Image Classification

The high adoption of mobile and digital cameras has resulted in a massive amount of multimedia content generated online, with images being a significant share of this data. On social media, billions of images are shared or posted daily [Per18], and analysing such a large volume of data is extensively based on machine learning algorithms. Image classification is one of the most-known tasks in the computer vision field, dealing with the task of labelling images with one or more predefined classes. Due to the ability to learn from a large amount of data, deep learning has become the standard computing paradigm in the field of machine learning, especially in the image analysis domain. In the following, we describe the main concepts behind the image classification task and introduce deep learning techniques. These are used later in the experiments conducted in Section 6.

2.3.1 Image Preprocessing

Preprocessing is the first step in the image classification task. This process distorts the raw data to make the data processing more efficient and leads to better feature detection later. There are various steps and techniques that can be performed in the preprocessing task. Some examples are listed here:

- image format and resizing – many datasets have images of different formats that vary in size. Considering that nowadays digital cameras generate images with high resolution, the processing time for these images is high. Therefore, at first, images should be converted into the same format and resized to a common size and aspect ratio (e.g., 224×224). Resizing the images is a critical preprocessing step, as it makes the models train faster on smaller images.
- normalize the data – the image RGB channel values are in the range of [0, 255], however, this range is large and not ideal for training the models. Therefore, the normalization step converts the values in the range for instance between [0, 1] by simply dividing by 255. In a neural network, a *normalization* layer could be added to rescale the values.

– data augmentation – is the process of altering the original images and generating new images. The goal of data augmentation is to increase the diversity of images by generating additional training data for the model without collecting new data. This process helps the model to generalize better and avoid overfitting when the model learns from a few training examples and causes negative performance on new examples. Examples of augmentation are changing the *orientation*, *rotation*, *cropping*, *histogram equalization* etc.

2.3.2 Feature Extraction

Feature extraction is one of the most important steps in a classification task. It is the process of transforming and usually reducing the dimensionality of the raw input data into more manageable and summarized information representing the object to be analysed. Feature extraction can be done manually or automatically [GGN+08]:

– manual feature extraction requires understanding the meaningful features for a particular classification task and implementing methods to extract those features. In many cases, it requires domain knowledge and an understanding of which features accurately represent the decision. This is a tedious process that is part of preprocessing and is often done in text mining scenarios (e.g., NLP), time-series analysis, and audio signal analysis, which require crafted features.

– automated feature extraction does not require human involvement, as this is done by specialized tools that automatically construct features from the raw data. This technique is very common for image analysis scenarios. Here, the automated feature extraction is part of the neural network and is done usually by the first layer. Automatic feature extraction techniques accelerate the end-to-end development of machine learning pipelines.

By transforming the raw data, feature extraction techniques have a critical impact on the classification task, and it requires the attention of not losing information from source data. In classical image classification approaches, new features are derived from pixel data such as colour histograms, shapes, and textures. However, such approaches become difficult to manage due to the many inputs that need to be tuned. For example, in a task involving the classification of animals, accurately determining the importance of colours or shapes poses a challenge due to the need for precise adjustments in these features.

2.3.3 Deep Learning Based Approaches

Though the theory behind neural networks was developed some decades ago, their popularity has massively grown in recent years owing to the advancement in the technology of computing power, which has consequently enabled the generation of large data as well. Consequently, the application of deep learning as a subfield of machine learning has become in demand. Its application is very common in image classification tasks due to the efficiency of processing a large amount of image data. A deep learning algorithm is basically a neural network with at least three layers: input, hidden, and output layer. It allows employing deep and complex architecture in neural networks. Deep neural networks are inspired and attempt to mimic the function of the human brain.

In the following, we introduce the main concepts behind the two types of neural networks that have been used in one of the conducted experiments in Section 6.3.

2.3.3.1 Multi-Layer Perceptron (MLP)

A Multi-Layer Perceptron (MLP) is the most basic deep neural network composed of multiple fully connected layers. Its fundamental concept is *perceptron*, illustrated in Figure 2.5. Considering a set of input features $X = \{ x_1, x_2..., x_{n-1}, x_n \}$ and their corresponding weights vector $W = \{ w_1, w_2..., w_{n-1}, w_n \}$, the perceptron computes the output y through the function given in Equation 2.4, where b is the bias.

$$y = W^T x + b \tag{2.4}$$

The perceptron algorithm named also as a single-layer perceptron, is the simplest feedforward neural network with a single neuron which is limited as a linear (binary) classifier, capable of learning linearly separable patterns. In a perceptron, the activation function is a step function (Equation 2.5). However, the perceptron is part of more complex models such as MLP.

$$f(x) = \begin{cases} 1 & \text{if } y > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.5}$$

Adding more layers and neurons allows for addressing more complex problems by neural networks. MLP is a feedforward network consisting of an *input* layer, one or more *hidden* layers, and an *output* layer. Each layer consists of

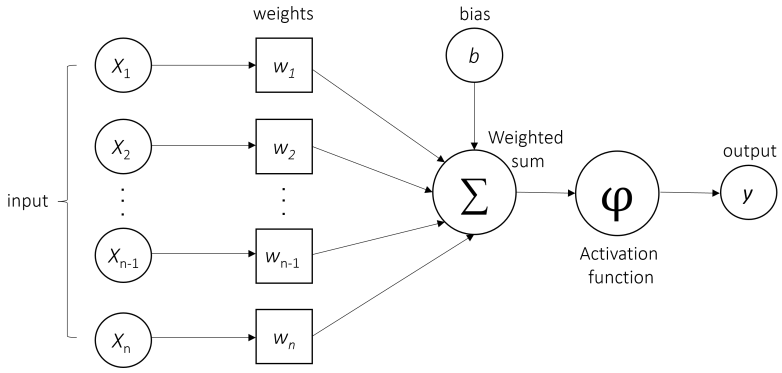


Figure 2.5 Illustration of the perceptron: the weighted sum plus the bias go through the activation function and generate the output

multiple nodes or neurons. The full connectivity indicates that each layer's output is the input of the next connected layer. Given the set of input features $X = \{x_1, x_2..., x_{n-1}, x_n\}$ where n is the number of dimensions, and a target label y from a finite set of k labels $Y = \{y_1, y_2, ...y_k\}$, MLP can learn a non-linear function approximator for classification. The input layer consists of neurons that represent features from the input sample data. Each neuron in the hidden layer converts the input values from the previous layer with a weighted linear summation $w_1x_1 + w_2x_2... + w_{n-1}x_{n-1} + w_nx_n$ followed by a non-linear activation function and connects to each node of the following layer. The non-linear functions allow addressing more complex problems beyond linearly separable patterns. Figure 2.6 illustrates the MLP. The neurons in the output layer generate values that are probability estimates (e.g., using the Cross-Entropy loss function) that the input sample belongs to each class y from Y. There are several non-linear activation functions, here we list the most used ones:

– **Sigmoid** – is a logistic function (Equation 2.6) that generates output values between 0 and 1, where a threshold of 0.5 assigns the input data to the positive class for values equal or above 0.5, and to the negative class if lower than 0.5. It is generally used for binary classification.

$$\sigma(x) = \frac{1}{(1 + e^{-x})} \tag{2.6}$$

– **Softmax** – the output of the softmax function (Equation 2.7) is a probability distribution, where the values are between 0 and 1, and they sum up to 1. These indicate the probability estimates of the input belonging to each of the target classes. Softmax is suitable for multi-class classification tasks with N classes, and the output vector contains the probability estimates

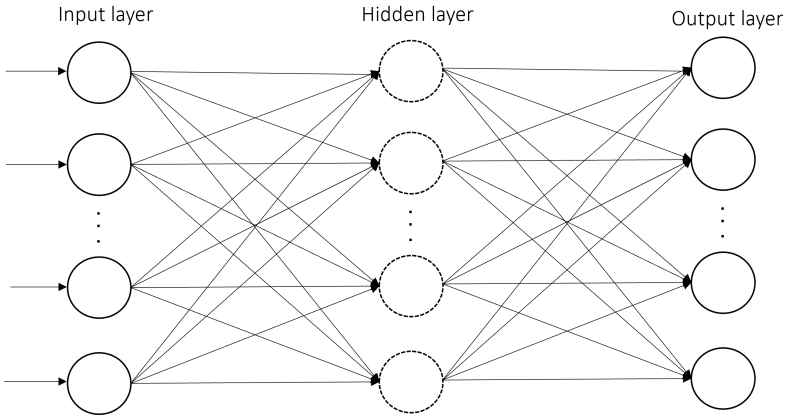


Figure 2.6 Illustration of a MLP, consisting of three layers: an input layer, one hidden layer, and an output layer, and each layer has multiple nodes.

that the example belongs to each class. The highest probability determines the "winning" class.

$$softmax(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{2.7}$$

– **ReLU** (Rectified Linear Unit) is a simpler and more efficient activation function (Equation 2.8) that allows models to learn faster. If the input value x is negative, it returns 0, otherwise the value.

$$f(x) = max(0, x) \tag{2.8}$$

In contrast to a perceptron, MLP supports multi-class classification, where the output can be a label from a set of more than two labels. The “winning” label is the label with the highest estimated probability. Furthermore, MLP supports multi-label classification where the input sample can belong to more than one label, and this scenario has been used in our experiment reported in Section 6.3.

Choosing the number of hidden layers as well as the number of neurons in each layer requires attention. If a neural network has too many neurons in a hidden layer, it can cause *overfitting*, memorizing the input patterns, whereas, too few neurons may result in a low representation of the input space patterns.

2.3.3.2 Convolutional Neural Networks (CNN)

Convolutional Neural Network (CNN or ConvNet) is a class of neural networks widely used for computer vision tasks such as image classification, image segmentation, object recognition [LBD+89], etc. CNN architectures distinguish from other neural networks as they explicitly assume images as input, and they learn

directly from image data, avoiding manual feature extraction. CNNs are similar to standard neural networks, consisting of layers and neurons that have weights and biases. Still, unlike regular neural networks, the neurons are arranged in three dimensions: width, height, and depth. A ConvNet architecture captures the spatial and temporal characteristics in images by applying relevant learnable *filters* or *kernels*. A ConvNet architecture consists primarily of three main types of layers:

1. **Convolutional layer** is the first and core of a CNN, where most of the computation happens. Considering that the input data is a coloured image, the input is made up of a 3D matrix of pixels, where the depth corresponds to the RGB values of the pixels. This layer has the kernel or filter which is spatially small but extends by moving over the receptive fields of the image, generating the output. *Stride* indicates the number of pixels that the filter window moves over the input image. The higher the stride number is, the smaller the generated output is.

2. **Pooling layer** performs down-sampling, reducing the spatial dimensions of the input volume, hence reducing the parameters and computation in the network. There are several pooling functions, among which *max* or *average* pooling being the most common. For instance, a pooling layer with a filter of 2×2 size and stride length of 2, will reduce the input image volume by 75% of the original size (Figure 2.8).

3. **Fully-connected (FC) layer** is the last layer that performs the classification based on the extracted features by previous layers. As in regular neural networks, the name "fully connected" indicates that each node in this layer is connected directly to all activations in the previous layer.

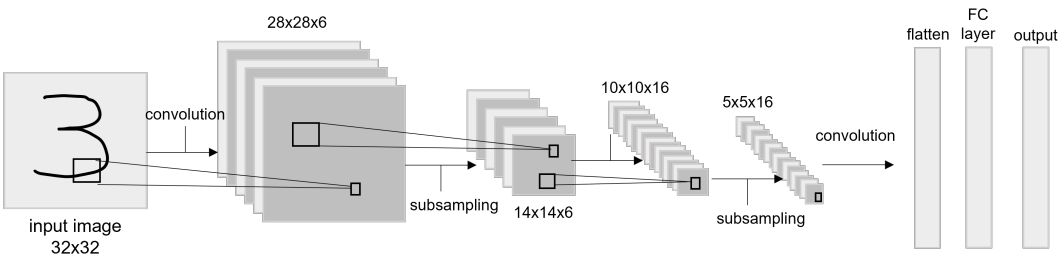


Figure 2.7 The LeNet-5 Architecture [LBB+98] - A CNN sequence to classify handwritten digits

A CNN architecture that consists of multiple layers transforms the original image pixels layer by layer into final class scores. The convolutional (CONV)

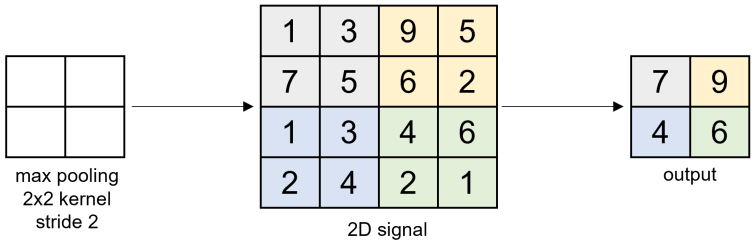


Figure 2.8 Max pooling - the output of the 2x2 filter is the highest element value in the filter window

and fully-connected (FC) layers, besides the activation functions, these layers perform additional parametrization (weights and biases of the neurons) which are trained with gradient descent where class scores are consistent with the training set image labels. Figure 2.7 illustrates an example of a CNN, the LeNet-5 architecture [LBB+98]. This network consists of seven layers in total, three convolutional layers combined with two pooling layers, and two fully connected layers. This multi-layer CNN is a simple and efficient architecture used for online handwriting recognition.

Advancements in the deep learning research area have accelerated the development of different architectures in the field of CNNs, such as AlexNet [KSH12], VGGNet [DDS+09], ResNet [SIV+17] etc.

2.3.4 Transfer Learning

Deep learning models usually require large datasets and heavy computational resources for training. On the other hand, in specific scenarios, there is a lack of labelled data, and access to labelling services is limited. *Transfer learning* is a technique that attempts to address both problems. Instead of training a model from scratch when large labelled data is available, or in situations where insufficient labelled data exist, the transfer learning technique utilizes an already trained model on different data and re-purposes the model to solve the new target task [WKW16].

There are two major transfer learning strategies that can be followed:

– pre-trained model utilized as a fixed feature extractor – in this strategy, a ConvNet model pre-trained on ImageNet [DDS+09] is taken and the very last fully-connected (FC) layer is removed while the entire network with its computed weights remains frozen. Then, a linear classifier can be trained for the new dataset. This strategy is recommended for scenarios when

the new dataset is small and is similar to the original dataset (pre-trained model).

– fine-tune pre-trained model – in this strategy, not only the last layer is removed, but also part of the network is to unfreeze, so the weights are fine-tuned by continuing the backpropagation. Depending on the scenario, it is possible to fine-tune only the last layers which become more task-specific, whereas the earlier layers remain fixed as they contain more generic features. This strategy is helpful for scenarios when the new dataset is small and different from the original dataset, on top of which a pre-trained model is trained.

In cases when the new dataset is large enough and labelled but very different from an original pre-trained model, it makes sense to train a CNN from scratch. In Section 6.3, a transfer learning technique with fine-tuning a pre-trained model is utilized. Results show that this technique is beneficial for scenarios when the new dataset is different from the original dataset, but the size of the dataset is relatively large and access to labelling the data is limited.

2.4 Performance Metrics

In a learning task, many metrics may be used to report the performance of classification algorithms, including *accuracy*, *sensitivity*, *precision*, *recall*, *f1-score*. Accuracy is defined as the number of correctly classified samples over the total number of samples. Accuracy provides a good indication of the overall performance of the system, however, it is not the proper metric to use when the dataset has unbalanced classes. For instance, an unbalanced dataset would be one that with two classes, e.g., real news and false news, and the aim of the system is to detect the false news. The two classes are not equally represented in the dataset. There would be more real news than false news (70% real vs 30% false). If a classifier would label each news story as real without examining the data itself, the performance of the algorithm would report a high accuracy of 70%. But this system would never detect and report any news as false. Therefore, other metrics that consider and report the classification performance based on detecting both positive (false) and negative (real) cases are appropriate.

In order to have a better picture of the performance of the system, there are four basic concepts to be considered: *True Positives* (TP), *False Positives* (FP), *False Negatives* (FN), and *True Negatives* (TN). TP occurs when a sample is *correctly*

predicted by the classifier which belongs to the *positive* class (e.g., the sample is false news and the classifier states it is a false news). On the other hand, TN occur when the sample is *correctly* predicted, and it belongs to the *negative* class (e.g., the system detects correctly a real news). FP are the cases when a sample is wrongly classified as positive (e.g., real news is classified as false news), while FN are the cases when samples are *incorrectly* classified as *negative* classes (e.g., the sample is false news and the system labels as real news).

The description and mathematical formulation of several metrics are reported below. All these metrics reach the best value at 1 and the worst score at 0.

Accuracy: As described above, accuracy calculates the ratio between the correctly classified samples compared to the total number of samples, see Equation 2.9. This is a valid metric to be used in scenarios when the classes in the dataset are well-balanced.

$$Accuracy = \frac{(TP + TN)}{(TP + FN + FP + TN)} \tag{2.9}$$

Sensitivity: Sensitivity or recall, is a measure of the accuracy of the classifier in detecting the true positives correctly, Equation 2.10. For the false news detection scenario, sensitivity refers to the number of false news that are correctly detected. Sensitivity is also known as *recall* and hit rate.

$$Sensitivity = \frac{TP}{(TP + FN)} \tag{2.10}$$

Precision: The precision metric measures the proportion of positive samples that are correctly classified, see Equation 2.11. For the false news detection scenario, precision refers to the number of false news that are correctly detected from the total number of false news predictions (including false positives).

$$Precision = \frac{TP}{(TP + FP)} \tag{2.11}$$

F1-Score: As a measure of a system's accuracy, the F1-Score is the harmonic mean of *precision* and *recall*, see Equation 2.12. This metric is a suitable metric to report on unbalanced datasets.

$$F_1 = 2 * \frac{precision \times recall}{precision + recall} = \frac{2 * TP}{2 \times TP + FN + FP} \tag{2.12}$$

The above-reported metrics are suitable for *binary* classification where the samples are assigned to one of the two classes (e.g., real vs false news), as well as for *multi-class* tasks where the data samples are assigned to only one of the

Table 2.2 Performance of multi-label classification metrics by example

#	Predicted labels	True labels	Accuracy	Zero-One Loss	Hamming-Loss
1	[1, 0, 0]	[1, 1, 0]	0	1	0.33
2	[1, 0, 1]	[1, 0, 1]	1	0	0
3	[0, 0, 1]	[0, 1, 1]	0	1	0.33
4	[0, 1, 1]	[0, 1, 0]	0	1	0.33
Overall performance			0.25	0.75	0.25

classes. However, there are classification tasks where multiple classes are associated with each dataset sample. This is known as *multi-label* classification, where a given set of input-output pairs $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, the objective is to learn a function $f : X \rightarrow Y$ that maps inputs X to outputs Y, where the corresponding sample label $y_i \in \mathcal{Y} = [0,1]^M$ with M labels. For instance, if we consider the example with the dataset of historical images, the task is to categorize each image into multiple classes. The list of classes is predefined and images can be assigned to one or multiple of the available classes: *place, person, object, event, or tradition*. There are several metrics suitable for multi-label classification which are described, and mathematical formulation is reported below.

Exact Match Ratio and Zero-One Loss: The Exact Match Ratio (EMR) corresponds to the *strict accuracy* where the entire set of labels must be correctly predicted, otherwise, the predicted sample is incorrect. Even if the predicted labels are partially correct, the score will be 0, see Equation 2.13. Zero-One Loss is the loss function of the accuracy, basically $Zero - OneLoss = 1 - EMR$.

$$EMR = \frac{1}{N} \sum_{i=1}^{N} \mathcal{I} \left[y^{(i)} \neq \hat{y}^{(i)} \right] \tag{2.13}$$

Hamming-Loss: Hamming-Loss (HL) is the fraction of the wrong labels to the total number of labels, see Equation 2.14. HL is a more suitable metric for multi-label classification tasks, as it does consider the predicted labels that are partially correct. This metric makes even more sense when the set of labels is bigger. For example, in the scenario with image classification with five labels, EMR would report very low accuracy, whereas HL score would be more meaningful.

$$HammingLoss = \frac{1}{N * L} \sum_{i=1}^{N} \sum_{j=1}^{L} \mathcal{I} \left[y_j^{(i)} \neq \hat{y}_j^{(i)} \right] \tag{2.14}$$

3

Profile Aware Crowdsourcing

Nowadays, there is a vast amount of generated data that can generate insights for organizations in general. However, this would be possible if there were comprehensive data processing tools and mechanisms for interpreting and extracting information from huge unstructured data. Crowdsourcing as a service leverages the intelligence of available online human participants, and it has helped organizations to carry out tasks that are not yet possible to complete by fully automated techniques [NK20]. The interest of the research community and the investment of organizations in the field of crowdsourcing has resulted in the great popularity of online platforms benefiting from crowd workers [GGM+18].

In the field of data processing, organizations have seen a significant advantage of utilizing online platforms such as Amazon Mechanical Turk (AMT), CrowdFlower, UpWork, or MicroWorkers, for various tasks such as information extraction, content moderation, sentiment analysis, entity resolution, multimedia processing etc. Crowdsourcing has great potential considering that as a field it intersects with diverse research fields, from social sciences to computer science [MP15]. The effort in the research area of crowdsourcing has advanced in both algorithms and systems directions, addressing the major challenges concerning quality, motivation, cost, and latency. Results obtained from crowdsourcing services can be of low *quality* due to many factors (difficulty of the task, payment rate, malicious crowd workers, lack of instructions, etc.). On the other hand, this service can be *slower* than expected (targeting wrong audiences, insufficient payments, etc.). Additionally, a posted job on a crowdsourcing marketplace can have a low hit rate even if monetary incentives are fulfilled, mainly due to *motivation* issue (complexity of the task, lack of gamification, etc.) Furthermore, for large amounts of data to be processed, it becomes relatively *costly*. These major challenges are correlated with each other, therefore optimal solutions are

a trade-off.

This chapter provides the foundations of the crowdsourcing topic, the concepts, and the major challenges and issues faced in this field. Crowdsourcing is the major component of the hybrid human-machine models we develop in this thesis, where the focus is on optimizing the trade-off triangle: data quality, cost, and latency. We describe two different crowd worker profile-based methods that contribute towards efficient large-scale data processing: i) models that consider the reputation and knowledge of the crowd-workers, ii) spatial crowdsourcing where the main focus is on the location aspect of crowd participants.

3.1 Concepts and Methodology

Crowdsourcing as a term was first coined by Jeff Howe, who defined it as the process of outsourcing a piece of work to a larger group of people via an open call for contributions [How06]. But, as a concept, it was introduced earlier by Luis Von Ahn [vAh05] who defined it as *"a paradigm that utilizes human processing power to solve problems that computer cannot yet solve"*. His contributions, such as the image-labelling game [AD04a], reCaptcha [VAMM+08], "Games with a Purpose" (GWAP) [AD08] are key to the crowdsourcing field. Since then, many successful applications that utilize crowdsourcing services have been implemented to solve computational hard tasks.

In order to solve a task or job, there are three main actors involved: a *requester*, the crowdsourcing platform *service*, and the *contributors*. The requester can be one or multiple people who have tasks to be solved, and they submit a job with tasks to the crowdsourcing platform, which is a service connecting the requesters with registered crowd workers who are ready to contribute to solving the posted tasks. This process is illustrated in Figure 3.1.

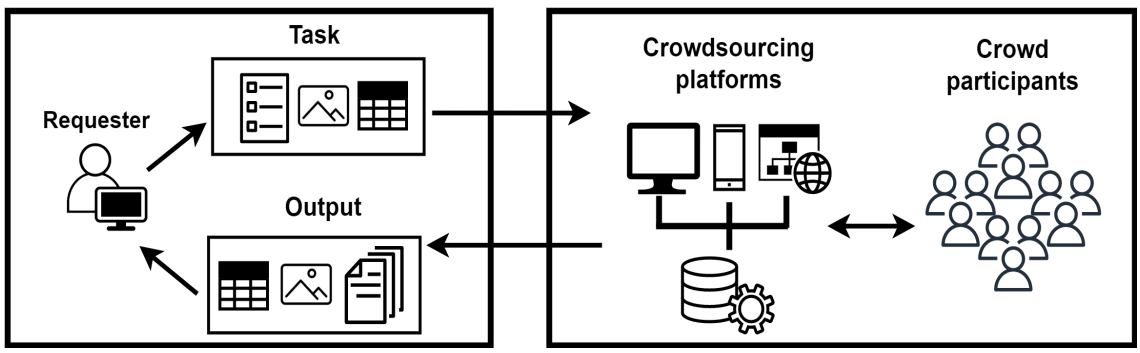


Figure 3.1 Crowdsourcing job workflow - Interaction between requester and contributors

While crowdsourcing as a topic has seen an increase in popularity, the number of crowdsourcing marketplaces has increased as well. There are more than 90 online platforms [MTD16] offering crowdsourcing services that are grouped into *macrotasks* and *microtasks*. Examples of macrotasks include translation of text documents, video and audio transcription, software testing, taxonomy creation, itinerary planning, etc. Micro-task crowdsourcing as a more popular service includes small labelling tasks such as image annotation and categorization, entity resolution, sentiment analysis, web content moderation, etc. The majority of online platforms are *paid crowdsourcing* platforms where online participants get paid for the completed tasks. In addition, there are several *volunteering* or *game-based* crowdsourcing platforms that offer additional mechanisms as incentives for completing tasks and which are not paid. There are several strong incentives behind *volunteer* crowdsourcing such as worthy cause, social reputation, entertainment, learning, or philanthropy. For instance, crowd workers solve tasks while playing games [AD08], or learning a new language [vAh13]. In many non-paid projects, users would answer crowdsourcing questions for the social good, such as the GalaxyZoo project [RBG+09] where participants volunteer to identify galaxies in astronomical photographs, the flood monitoring and detection project where crowd users share photographs to determine flooded streets [WSWA+19], or the example during the Haiti Earthquake where volunteers were asked to translate tweets [ZGS+10].

So far, we have mentioned a few terms that are part of the crowdsourcing process, and here we will list the main terms that will be used throughout this thesis:

- **Requester** – the person or team designing the task to be completed by crowd workers/contributors
- **Crowd Worker/Contributor** – the person performing the task completion. The terms crowd worker and crowd contributor will be used interchangeably.
- **Crowdsourcing Platform** – the online web or mobile-based platform that connects requesters and crowd contributors, manages the submitted task and collects the contributors' results.
- **Task Design** – steps that involve generating the description of the task, the detailed instructions for the crowd workers on how to complete the tasks, and additional configurations about inclusion and/or exclusion criteria applied to crowd workers.

- **Task Assignment** – this is the process of how the tasks are assigned to contributors. Several approaches exist: a requester can on demand assign the tasks to a target group of contributors, or the tasks can be assigned automatically based on algorithm setup.

- **Micro-task** – this is the most popular task in crowdsourcing settings. The task is small, short, and precisely defined, usually with closed answers (e.g., single or multiple choice). Microtasks are also known as Human Intelligence Tasks (HITs).

- **Macro-task** – is a bigger problem that requires more time to be solved (e.g., audio or video transcription, translation of a text document).

- **Crowdsourcing job** – a small project or a list of tasks to be crowdsourced (e.g., annotate a collection of images).

- **Reward** – the incentive for the contributors, which is usually monetary (paid crowdsourcing), or non-monetary compensation (volunteer crowdsourcing).

- **Cost** – the monetary cost to execute a crowdsourcing job.

- **Latency** – the elapsed time for a task to be completed.

- **Answer Aggregation** – usually the same task is assigned to multiple contributors and the answers are aggregated with the aim of increasing the quality of obtained results.

- **Worker Quality/Accuracy** – this is the score that indicates the percentage of provided good quality answers from total answers.

- **Worker Reputation** – this is the score that is estimated based on a worker's history of completed tasks (cumulative accuracy).

There are two crowdsourcing models with respect to task assignment:

- *pull crowdsourcing* – this is the most popular approach, where crowd contributors are logged in to the platform and search for tasks to be completed. They can navigate through available jobs and choose which one to contribute.

– *push crowdsourcing* – contrary to the pull model, this method will push tasks to a target group of contributors. Usually, the contributors are chosen based on the analysis of their profiles to best match the tasks with contributors.

3.2 Challenges and Issues in Data Crowdsourcing

When referring to crowdsourcing, this thesis addresses data crowdsourcing problems. In the following, we will list and describe the main challenges that are present in crowdsourcing research: *data quality*, *cost*, *latency*, and additionally *motivation* that are applicable to volunteer crowdsourcing scenarios.

3.2.1 Data Quality

Crowdsourcing tasks are solved by humans and therefore prone to error. Crowdsourcing may yield relatively low-quality results if not properly designed and managed. We distinguish two rationales leading to this: mistakes due to job requesters and mistakes due to contributors.

Low-quality results can occur due to mistakes done by *requesters*. Here we list some scenarios that lead to this:

– lack of clear or misleading *instructions* – when designing a job, crowdsourcing platforms emphasize the importance of instructions, and as a countermechanism, they have implemented many pre-configured templates.

– unfair *payment rate* – this results in low interest and hit rate, limiting the job to a smaller audience of contributors with less experience. Due to the low pay rate, contributors can also not pay attention when answering a task in order to maximize the number of tasks. Crowdsourcing platforms have configured limits depending on the task and would remind and suggest the proper payment rates.

– *task complexity* – if the task is designed in a way that it contains multiple microtasks, it is better to decompose the task. Higher task complexity has been shown to decrease the accuracy of results. [TCZ+18].

– wrong *task assignment* strategy – depending on the difficulty and context of the task, the crowdsourcing job might require finding a suitable crowd worker group. The strategy would be to consider the profile aspect of contributors: location, qualifications, reputation, etc.

On the other hand, obtaining low-quality results could be due to *contributors*, and this scenario is more critical. Here we list some reasons that can lead to this:

- level of *expertise* – crowd contributors have different levels of expertise, and some would be *trained* for specific tasks. An untrained worker has higher chances of not accomplishing tasks that require specific *skills*. In some scenarios, task design techniques such as clear instructions and guidelines with proper examples of how to solve the task can help to address this issue. Additionally, task assignment techniques that leverage the profile of contributors are more efficient in matching the task with the expertise of contributors.
- *dishonest or malicious* workers – there are cases when crowd workers would intentionally provide wrong answers. Usually, these workers tend to submit random answers with the aim of maximizing their earnings. There are several well-known quality mechanisms addressing malicious workers. *Gold questions* are a set of questions with known answers that are used as a *qualification test* for workers before performing the task. *Programmatic gold questions* [OSL+11] are ad-hoc or systematic qualification tasks that are injected during the crowdsourcing task as well.

Data quality control is a major challenge in paid crowdsourcing that has drawn attention. As a result, several quality control mechanisms have been proposed. In Section 3.4 we detail the quality control techniques that are used in our hybrid human-machine models and the experiments.

3.2.2 Cost

In paid crowdsourcing, participants' contributions are not free. The price for a micro-task ranges between $0.01 to $10.00 [Ipe10], and 90% of the HITs have a rate lower than $0.10. Though the price per HIT is negligible, crowdsourcing becomes expensive when dealing with a large number of tasks.

Example 3.1 An example of entity resolution problem

A travel agency provides hotel booking services online. This company obtains hotel data from different data sources, and it needs to aggregate the data into a clean database. For instance, they have obtained two databases about hotels in Europe that have 4'000 and 2'000 records respectively. The challenge is that the same hotel can appear in both databases with different levels of information,

therefore, the agency needs to merge the duplicate records. Finding records that refer to the same real-world entity across different databases is known as deduplication or entity resolution.

For instance, if we consider the entity resolution problem presented in Example 3.1, a naive method would consider the full pairwise comparison between the records in the two databases results in a huge number of tasks ($4000x2000 = 8M$). Solving this problem becomes very costly, even if the tasks have a low price. For illustration, applying purely crowdsourcing for eight million HITs and assuming the cost per HIT is \$0.01, the minimum cost would be \$80K. Considering that quality control mechanisms would be necessary, the cost increases further. Furthermore, if qualification tests and injected test questions are applied, the cost would increase by at least 10%. Furthermore, applying a redundancy mechanism to maintain high-quality data, e.g., each task is solved by three or five different crowd workers, the cost would increase by 30% or 50%. Additionally, bonus credits could be used to boost the motivation for participation and data quality, and that adds to the overall cost. As illustrated in the example above, crowdsourcing larger datasets becomes expensive and requires cost-control mechanisms.

3.2.3 Latency

We have seen that crowdsourcing large datasets can get expensive, but completion takes time as well. *Task duration* is the time to complete a task by a crowd worker. Latency refers to the time needed to complete all tasks in a crowdsourced job or project. The entity resolution problem presented in Example 3.1 can take hours or even days to complete. Latency is a very important factor when considering crowdsourcing. Several reasons can lead to high latency in crowdsourced jobs. An important factor is the *task reward*. If the task price does not reflect the task complexity, that will cause low participation. *Task difficulty* is another reason that can make tasks unattractive for contributors. Unclear task instructions and lack of task design mechanisms can lead to higher *task complexity*, and that impacts the responsiveness of crowd workers. Latency depends as well on the crowdsourcing platform itself. More popular platforms have numerous registered and active crowd workers, offering high availability and participation. Depending on the demographics of platform members, the time when the crowdsourcing job is launched matters. It is necessary to con-

sider this in cases when the task is filtered for specific regions. Additionally, quality control mechanisms such as redundancy and complex task assignment strategies can lead to higher completion times.

3.2.4 Motivation

Finding the right incentives to motivate participants is challenging. Factors that influence the users to participate in crowdsourcing projects are either *extrinsic* motivations or *intrinsic* motivations.

Extrinsic motivations are mainly monetary rewards, and paid crowdsourcing platforms are the best examples of extrinsic motivations, where platform participants get paid for solving simple tasks. Extrinsic motivations can be non-monetary as well, where users participate to share their knowledge, enhance their personal or professional skills, or gain social reputation and compete within a participatory user community [ALSG17].

Intrinsic motivations are connected to self-determination and self-esteem, without expecting a reward for the contribution. Based on the self-determination theory [CLP+18], intrinsic motivations to volunteer are strengthened with the presence of a social cause [RBG+09; ZGS+10]. In these scenarios, crowd users are intrinsically motivated to participate by enjoying contributing to the project. The main intrinsic motivations are *satisfaction*, *fun*, and *learning*. This has led to crowdsourcing projects that implement game-based approaches, named *gamification*. Elements of gamification are *leaderboards* where crowd users get *badges* and *points*, competing within the user community [MHK16].

While extrinsic incentives motivate participation with monetary rewards in paid crowdsourcing, the combination of intrinsic incentives and gamification is key to the success of projects in voluntary crowdsourcing initiatives.

3.3 Gamification in Crowdsourcing

Crowdsourcing relies on the contribution of online participants who are motivated to solve tasks for a small monetary reward. On the other hand, crowdsourcing tasks are *gamified* to make them as enjoyable and attractive as playing a game. Von Ahn and Dabbish [AD08] announced the concept of *games with a purpose (GWAPs)*, where people can collectively solve computational problems by playing online games.

The ESP Game [AD04b] is a famous GWAP that pairs randomly two online players, to whom a random image is shown. Players collaboratively tag the images, by competing on guessing the tags provided by the game opponent. Peekabom [ALB06] is another GWAP that improves the ESP game by providing precise information about the location of the objects in the images. This game has led to the creation of large datasets that are needed for training computer vision algorithms. Since then, many gamification crowdsourcing projects have been successfully implemented. For instance, the FoldIt [CKT+10] is a revolutionary crowdsourcing project where online players "fold" the structure of a virtual protein, and contribute to the challenging scientific issue of protein folding. Duolingo [vAh13] is a crowdsourcing application where players contribute to translating pieces of documents while learning a language.

Gamification is a crucial technique to increase the motivation for participation in voluntary crowdsourcing. Studies have shown that gamification has a positive impact on crowdsourcing projects [MHK16]. Positive effects are that participants show an increase of engagement in the long-term, higher quality of obtained results, and in contrast to paid crowdsourcing tasks, there is less intention to cheat. The main gamification elements recommended are scoring and ranking. Crowd participants get *points* as a reward for completing micro-tasks, and *leaderboards* motivate competition to get a higher reputation within a social community. *Badges* are additional motivation as they level the reputation.

3.4 Data Quality Control Mechanisms

Crowdsourcing as a methodology allows data annotation at scale, and data quality is the key aspect of this process. If not properly managed, a crowdsourcing job can lead to low data quality. Quality control is one of the major issues in crowdsourcing, and in Section 3.2.1, we have explained a few reasons that can cause this. Therefore, the quality control topic has attracted the focus of research communities to a great extent. In the following, we will describe well-known quality control methods that can be implemented in combination and help to obtain high-quality data. Such methods filter wrong answers from honest workers, as well as avoiding answers from workers who intend to compromise the quality of results. Figure 3.2 illustrates the fur main dimensions that characterize quality control in crowdsourcing systems: the i) worker's profile and ii) task design features [ABI+13], complemented with iii) task assignment methods, and iv) finally results aggregation techniques.

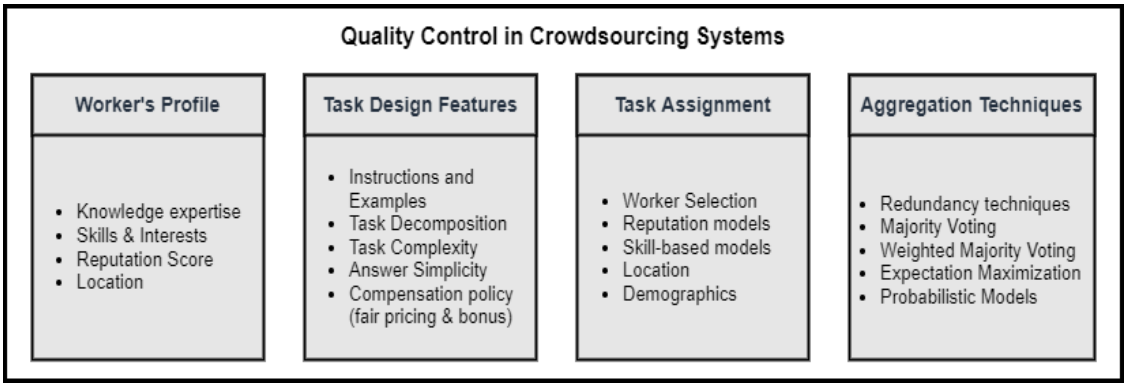


Figure 3.2 Taxonomy of data quality control mechanisms in crowdsourcing systems

3.4.1 Qualification-based Models

In order to assess if crowd users are willing to participate in a bigger task, the job requester can deploy qualification tests. Qualification tests include a set of questions for which the answers are known in advance. It is good practice to have a large enough set of questions so that they are shuffled and randomly assigned to different crowd participants. Based on the results from the test, crowd users are allowed or rejected to participate in the tasks.

3.4.1.1 Gold Questions

A well-known mechanism for filtering qualified crowd workers is the *gold questions* [OSL+11]. These are questions for which the job requester knows the answer, and these are used to assess the knowledge of task participants. It helps to filter competent users, as well as *fraudulent* users who might provide random answers with the aim of not being detected, increasing the task submission rate and reward. Only participants who achieve a certain number of correct answers from gold questions are allowed to participate in the crowdsourcing job. Online crowdsourcing platforms such as FigureEight[1] have implemented such qualification tests (i.e., quiz page) with at least 10 gold questions which are used at the beginning to test the competence of the worker. A crowd worker is considered trusted if he achieves at least 70% accuracy in the quiz mode, and this threshold is configurable by the job requester. Qualification tests can be implemented on other platforms such as MTurk or MicroWorkers, but they have to be manually created by the requester. The gold questions mechanism is fairly simple and fraudulent workers can overcome it by cooperating and easily spotting the

[1] *Now Appen https://www.appen.com

answers, hence evading the qualification test and entering the crowdsourcing real tasks afterward. Therefore, the set of gold questions should be larger and randomly shuffled among different users, shuffling the order of the answers for each question as well.

3.4.1.2 Injected Test Questions

Injected test questions are gold or verifiable questions that are injected into the task after the qualification test. The aim is to continuously assess the results of the crowd participants. Since qualification tests usually are transparent to the crowd workers, injected test questions are not. For instance, a good practice is to have 1 out of 10 questions as a gold question. This quality mechanism monitors the performance of workers constantly while solving tasks in a project. If the accuracy drops below a certain threshold (e.g., 70%), these workers are marked as inadequate and banned from that project, and their contributions are flagged as unacceptable and removed from the resulting output. While the qualification test models control the data quality, the disadvantage is the cost aspect, which increases. The crowd workers need to be paid when answering these questions too, as they are part of the project. Therefore, the size of the gold questions is a trade-off between cost and quality. The size of the gold questions should be optimal, should be non-ambiguous questions, and not be repeated, avoiding being recognized as gold questions. In case the size of gold questions is very limited, it is recommended to limit the number of tasks that can be solved by a single crowd worker up to a level when the gold questions used in the qualification test and injected test questions are consumed. For instance, if there are only 20 gold questions available, ten of these questions would be used for the quiz mode, and the other ten questions would be injected in one out of ten questions, i.e., ten out of 100 questions/tasks. For up to 100 tasks, gold questions would not be repeated, therefore, the number of tasks that can be solved by a single crowd worker would be limited to 100 in advance.

Qualification-based methods are meant and biased to eliminate unethical workers, however, they penalize honest but low-skilled workers not fitting for specific tasks too, discarding them from the labour pool. Adapting the gold questions [MGB15] to the worker's skills can help to identify honest low-skilled workers, allocating gold questions according to their skill set and allowing them to contribute in tasks fitting to their skills.

3.4.2 Task Assignment Models

Given a pool of tasks and a list of available workers, the task assignment models try to solve the problem of assigning the relevant tasks to adequate workers, such that the overall result quality is maximized and cost minimized. An ineffective crowdsourcing task assignment strategy can result in low-quality outcomes, leading to both a financial and time investment loss. Optimal task assignment models explore the workers' profiles and the nature of the task to find the right match between the tasks and available workers. Based on past studies, two major groups of task assignment strategies have explored workers' profiles: i) strategies that consider the worker's *reputation* and *knowledge*, and ii) *location* based strategies.

3.4.2.1 Reputation and Expertise Models

The quality of judgments provided by crowd workers depends on the knowledge and skills they possess. Tasks that require specific skills or expertise, have to additionally target the suitable worker group. Some crowdsourcing platforms (e.g., MicroWorkers or MTurk) provide access to profiles of their platform members, allowing a job requester to search for groups of workers with common skills or knowledge. However, the more specific the task is, the more limited searching options are available. Therefore, tailored strategies can be implemented by assessing the knowledge and interest of crowd participants to find the relevant workers. For instance, if a task requires knowledge about sports, arts, or history, crowd users with *knowledge and interests* in these topics would be fitting to this task. Workers need to demonstrate their expertise, and how capable they are at doing certain tasks. To prove their expertise, workers can provide *credentials* or *experience* [ABI+13; SSD12]. Credentials can be documents that workers submit to crowdsourcing platforms when completing their profile, proving their capabilities to job requesters. For instance, crowd workers can provide documents proving their education, language certificates, pieces of training, etc. On the other hand, experience is more about proving their knowledge and skills from the work they have done in crowdsourcing projects.

Crowd workers build their reputation scores while participating in projects. *Reputation* is a public and general metric that represents the trustworthiness of the worker, but the expertise is task-dependent. Therefore, these two aspects complement each other. Crowd workers' reputation scores are continuously evaluated on their contributions to crowdsourcing projects. This raises aware-

ness among workers to provide high-quality data and compete within their community. Exploring the worker's profile, such as the reputation, knowledge, and skills, can help in deriving the *reliability* of a worker, which goes beyond the reputation and is based on the context of the task problem too.

3.4.2.2 Location-aware Models

An important aspect when considering the profile of crowd workers is their location. The rapid growth and high adoption of mobile devices connected wirelessly have enabled the *spatial crowdsourcing* paradigm to solve ubiquitous problems. Spatial crowdsourcing enables efficient task assignment to users carrying a smartphone [ZYL+19]. Moreover, a great number of crowd users utilize social networks, making them crowd providers as well. For instance, crowdsourcing advantages have allowed crowd communities to contribute to emergency responses and disaster management [TRP21]. Active crowd users, who are geographically located near the areas of interest, can reach and share information faster than the mass media systems can. However, one of the main limitations is the data quality or trustworthiness of data providers. In parallel to considering the location of the users, location-based task assignment models need to assess the credibility of crowd participants. In addition, context-aware task allocation models examine other properties of the worker's profile, such as skills and interests. Full *context-aware* models utilize the three properties of crowd worker's profile: i) *location*, ii) *reputation*, and iii) *skills*, *interests*, and *expertise*.

Advances in mobile technology have enabled crowdsourcing of sensor data from users' mobile devices. Frequently, tasks related to mobile sensing, such as assessing ambient noise levels and monitoring air pollution, are outsourced. This has led to a new field of *crowd sensing* [WWW+18], where mobile users perform location-based sensing tasks. For instance, based on their location, crowd users are asked to report mobile sensor measurements.

3.4.3 Results Aggregation Methods

Paid crowdsourcing platforms attract a vast amount of online participants due to the monetary reward opportunities. These platforms are attractive for *spammers* that often submit random answers, without looking at the task, aiming to collect the respective fee upon task completion [GPGM12]. To maximize the reward, they can create bots submitting automated answers, acting as human workers solving tasks. While quality control mechanisms such as qualification

tests have shown as effective in filtering spammers, on the other hand, techniques to overcome such filters have advanced as well. *Malicious* workers can collaborate in groups to circumvent quality control mechanisms. As a result, further methods focusing on *task redundancy* [MCK+13] have been introduced. Redundancy techniques assign the same task to multiple workers and use aggregation methods to derive reliable results. Simple aggregation methods rely on voting strategies such as majority voting. More advanced algorithms consider further the worker's profile and behaviour while solving tasks in a project. In the following, we describe these two strategies.

3.4.3.1 Voting Techniques

The *majority voting* (MV) is a crowdsourcing results aggregation technique that can be applied to binary, multi-class, and multi-label classification problems. Very often, relying on a single worker can be unreliable. Statistically, the more workers assigned to a task, the higher the quality results are obtained since wrong judgments are eventually cancelled out from correct answers [KLW+03]. A simple approach to achieve results with a higher quality is the MV. MV is a redundant task strategy where the same task is assigned to multiple workers. This approach estimates the actual ground truth based on the "winning" label, which is the label with the highest number of votes. For instance, in a simple multi-class labelling task about sentiment analysis, where possible labels are "positive", "negative", and "neutral", and the task has been assigned to five workers, if three workers answer with a "positive", and two workers answer with a "neutral", based on the MV technique the final label will be "positive" (three out of five). In simple binary tasks, it is recommended to have an odd number of workers so that there will be always a majority vote. Best practices suggest to use three or five redundant annotators. The main drawback of MV is that it treats each worker's answer as equal in quality, i.e., the reliability score of each worker is considered the same. However, in reality, this is not the case.

Weighted majority voting (WMV) is an extended version of MV used to improve model performance and ideally achieve a higher output quality than simple majority voting [LY14]. The WMV considers assigned weights to each label and does a weighted sum when deriving the winning label. For instance, the weights can be assigned based on the worker's profile (e.g., reputation of a worker, competence level of the worker, etc.).

3.4.3.2 Dawid-Skene Model

The Dawid-Skene (DS) model is a voting strategy that takes into consideration the reliability or competencies of the annotators. This model was developed by Dawid and Skene in 1979 [DS79], with the initial goal of solving problems in the medical domain, such as combining different opinions of multiple physicians for medical diagnosis. Since then, the model has been improved and widely applied for classification tasks in different domains. DS model considers the error rate of the annotators to derive the "winning label". The DS model has been extensively used in crowdsourcing applications for aggregating crowd workers' results. Applying the redundancy method, asking multiple users to annotate the same task, the DS model estimates the *expertise* of the workers and infers the *true* label.

DS model is an iterative algorithm based on the Expectation-Maximization (EM) algorithm [DLR77] and it estimates worker error rates in three steps:

1. In the first step, the algorithm *initializes the correct label* for each task by applying the majority voting approach.

2. Second, it *estimates the error rates* for all annotators by considering the *correct labels* from *step 1*. This is known as *step E*.

3. And third, the algorithm *estimates the correct labels* by weighting the annotations from *step 1*. The weights are the estimated error rates of annotators from *step 2*. This is known as *step M*.

The algorithm iterates between steps E and M until convergence. In each iteration, a confusion matrix is calculated for each annotator. The rows in the matrix represent discrete conditional distributions of annotators' judgments given the ground truth answers for each class. The diagonal elements in the matrix represent the correct classification rates (accuracy), whereas the off-diagonal are misclassification rates for each class. The true labels are estimated based on the maximum likelihood principle, by jointly considering the provided annotators' labels and their confusion matrix.

There have been many efforts to improve the DS models, working on reliability and optimization of data aggregation in crowdsourcing. For instance, [RYZ+10] proposes a Bayesian approach to compute prior probabilities to each class, and the worker confusion matrix is randomly initialized. However, this model is limited to binary classification scenarios. An extension of the DS model

has been made by [BMG+12] which takes into account and models the task difficulties as well. However, the DS model remains a robust technique that is still considered a baseline for performance evaluation for new proposed methods.

3.5 Task Design Best Practices

Crowdsourcing can help data collection at scale, however, ensuring high-quality data is challenging. An important factor for success is task design, which is a difficult process. There are few studies [Alo13; MP15] that have researched the area of effective crowdsourcing with a focus on task design. These studies recommend instructions that, if properly implemented, can provide better results. In the following, we will overview a few aspects which need to be followed when designing a task.

3.5.1 Instructions and Examples

Unclear instructions often can confuse crowd workers and lead to a mismatch between worker understanding and task needs [GYB17]. Therefore, clear instructions describing the task to be solved are the first step in the task design process. Sometimes, the job requesters might miss the nuances of their tasks and provide ambiguous instructions. As a result, instructions need to be reviewed and often feedback from the crowd workers is required. A good practice is to provide concrete examples next to the instructions about how to solve the task. Additionally, it is useful to ask someone else to go through the instructions and examples and get their feedback if further review is needed.

3.5.2 Task Decomposition

Often crowdsourcing tasks can be complex too, hence, complex tasks must be decomposed into smaller and easier sub-tasks. Smaller sub-tasks can be executed in sequential where the crowd worker is asked to answer all sub-tasks, or in parallel, asking multiple crowd workers independently. Task decomposition helps to avoid long and complex instructions followed by too many examples. If a task contains multiple questions, good practice is to decompose it, creating a micro task for each question. For instance, an image labelling task might have two different questions. In some scenarios, task decomposition could be helpful for multi-label classification tasks in case the set of labels is large, and the task is difficult. Creating binary answer tasks for each label can simplify and make the

task easier to solve. Though decomposition leads to a higher number of tasks, it does not necessarily lead to higher costs, as the simpler tasks are expected to have lower HIT prices.

3.5.3 Simplify Task Answer

The task usually involves questions for which an answer is expected from crowd participants. The choice in the answer set should be close-ended, usually a single-choice question. It is recommended to avoid open-ended or free-text as that can lead to higher errors, and these questions are harder to answer. Multiple choice is acceptable if the task is simpler and clear instructions are provided. Otherwise, the task can be decomposed into single-choice questions. For instance, if a task involves image labelling, it is better to limit the number of potential answers rather than leaving the option to enter any kind of answer in a text box. The answer set can be defined and populated a priori.

3.5.4 Fair Pricing

Monetary reward is the main incentive in paid crowdsourcing scenarios. Therefore, assigning a fair price is critical for the quality of obtained results, as well as the time to complete a crowdsourcing job. In order to understand better what price is suitable for your task, it is recommended to try and solve a few tasks and log the timing required to complete them, hence estimating the hourly rate of tasks completed. As job requesters would be biased and have good knowledge of the problem, the best feedback would be obtained from someone who is not a subject expert. Some online crowdsourcing platforms would have pre-defined limits for different categories of tasks, giving an indication what is the expected fair price. Fair pricing means a higher pay rate than the minimum hourly wage. Furthermore, these platforms have the *bonus* price option as another incentive for motivating participation and boosting the data quality. In scenarios where latency is not critical, a strategy is to crowdsource a smaller chunk of tasks to analyse and obtain a list of suitable crowd workers that provide good quality answers, and later consider them for the full list of tasks and provide them bonuses for their good work. This strategy will keep them motivated and also raise their reputation, making their profile visible for other tasks.

3.5.5 Crowd Selection

Many online platforms are globally accessible and have active workers from different regions of the world, enabling income opportunities for anyone with an internet connection. Registered users differ by age, location, education, interest, and skills, offering opportunities to target specific demographics when necessary. Some tasks would require general knowledge and any participant would be suitable to participate. However, some tasks would require knowledge of specific topics, and education and skills matter. Furthermore, a task can cover topics to a specific region, country, or language, therefore location and language skills are important. Most platforms would allow job requesters different selection features, allowing them to assign their tasks to different regions (continents, or countries), based on spoken languages, education, etc. Furthermore, popular tasks have led to the formation of various groups centred around specific topics, for instance, *sentiment analysis*, *image annotation*, *image segmentation*, *image categorization*, *entity resolution* etc.

3.5.6 Manage Data Quality

Data collected from crowd users can be of low quality due to different reasons coming from the job requester such as unclear instructions and task high complexity, but also due to genuine or malicious errors from participants. Genuine errors occur when crowd workers provide wrong answers due to their lack of knowledge of the task or due to misunderstandings. On the other hand, malicious errors occur when participants provide random answers without paying attention to the questions. Malicious behaviour is done to increase the task submission rate and the reward. Therefore, proper data quality control mechanisms have to be implemented in order to manage the quality of crowdsourced data. Furthermore, malicious workers can cooperate in groups and coordinate their answers to bypass quality checks, therefore, multiple mechanisms at different stages need to be implemented. As discussed in Section 3.4, qualification tests are the first barrier to filtering incompetent and malicious crowd workers, in conjunction with the injected test questions approach. A redundancy mechanism with basic voting aggregation techniques can be considered for simpler tasks, whereas, more complex tasks require profile-based task assignment techniques.

4

Bridging Machine Learning and Crowdsourcing

In Chapter 2, we have described the concepts of *machine learning* topic and the benefits it brings to large data processing problems. However, we have seen that accuracy remains the main challenge in this field for many tasks (Section 2.1). On the other hand, in Chapter 3 we have introduced the concepts behind *crowdsourcing*, and we motivated the benefits of involving humans in solving tasks that remain difficult for machine learning algorithms. Nevertheless, we have seen that crowdsourcing alone is not scalable, as it is costly and slow (Section 3.2). Moreover, in Chapter 1, more specifically in Section 1.2, we have stated the above limitations that arise in these two fields, and we illustrated them with concrete examples. Considering the advantages and limitations of both machine learning and crowdsourcing approaches, the combination of both methods sheds light on overcoming these limitations. Therefore, the need for bridging machine learning and crowdsourcing is seen as the scalable solution to achieving higher accuracy at low cost and latency.

The combination of automated data-solving techniques such as machine learning and human-based methods such as crowdsourcing has led to a new research topic, *hybrid human-machine information systems* [DDG+17]. In this chapter, we provide the background behind this new and emerging research topic. Since this topic involves *information systems* field, we briefly recall what information systems are. Afterward, we elaborate on the hybrid human-machine information systems method. Further on, we present various fields that have benefited from the applications of these hybrid approaches.

4.1 Information Systems: Basic Concepts

People and organizations use information systems daily. In a high-level definition, an Information System (IS) is a set of interconnected components that *collect*, *process*, *store*, and *distribute* data and information [SR17]. An IS does not necessarily involve information technology, for instance, a simple system based on paper and pencil can be used to collect, process, and store data and information. Nowadays, the IS definition is more related to computer-based information systems (CBIS) which makes use of hardware, software, and communication technologies to process raw *data*, which is transformed into *information* and *knowledge*. Advancements in these technologies have enabled the processing of a large amount of data. Generated information and knowledge is disseminated for supporting *analysis*, *visualization*, and *decision-making* in organizations. In this work, with information systems, we refer to computer-based information systems.

As information systems involve people and technology in general, from the socio-technical perspective, there are four fundamental *components* in IS: *people*, *roles*, *tasks*, and *technology* [OWK99]. The *technical* part of the system includes technology and tasks. *Technology* can refer to any hardware, software, and communication networking tools used to perform *tasks* to achieve the organizational objectives. *People* as part of the system have *roles*, and they perform tasks as well, which eventually generate results from which the organization benefits. The modern approaches to IS consider additionally *data* and *processes* as very important components [SR17]. The *data* refers to a collection of raw facts, which need to be stored, aggregated, indexed, and organized into *databases* in order to be processed by other components, and generate information and knowledge. The *processes* are defined as a set of activities that need to be followed in order to achieve the desired output. Figure 4.1 illustrates the components of information systems.

The ultimate role of an information system is to help organizations in various aspects, e.g., improve their efficiency, automate routine tasks, assist in decision-making, facilitate communication and collaboration, etc [BSW+19]. This is achieved by transforming *data* into *information* and into *knowledge*. Data are raw facts that without further processing and analysis do not add value to organizations. Therefore, raw data must undergo a *transformation* process that involves several steps (data collection, organization, processing, integration, reporting, and analysis) to transform it into information [BSW+19]. If interpreted

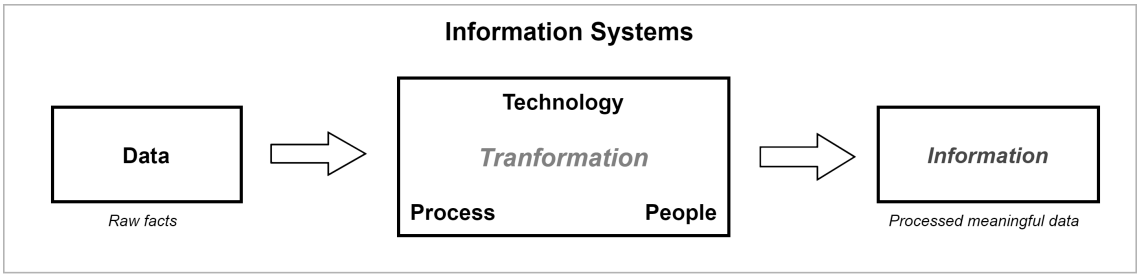


Figure 4.1 Information Systems Components [BSW+19]

correctly, the generated information is transformed into knowledge.

We use information systems in our daily lives, they are part of a wide range of fields, such as finance, education, human resources, e-commerce, healthcare, etc. Nowadays, the use cases and applications of information systems are countless. Here, we outline two examples that are linked with our scenarios described in Section 1.3. For instance, we use daily social media as a channel for communication. Social media are information systems that collect, process, store, and share data to facilitate social interaction and to keep people informed. Another example is cultural heritage institutions such as museums or digital archives. These institutions make use of different information systems to store, manage, organize, and make accessible their collections. The scenarios described in Section 1.3 are directly related to the later two examples (social media, cultural heritage institutions).

4.2 Hybrid Human-Machine Information Systems: A Preview

There have been substantial advances in the technology component of information systems, in all three dimensions: hardware, software, and communication tools. This has caused the generation of enormous data for organizations, therefore sophisticated data processing tools are needed to solve different data-related problems. For instance, organizations while they collect and store data from different sources, they can store in their databases redundant or different records that are related to the same concept or issue. As an example, a company maintaining a database about products might have different entries for the same product. This problem is known as entity resolution, and its task is to disambiguate records that correspond to entities across or within the same databases. A very popular and important task when collecting and storing data is data annotation. As an example, many companies running their business online need

to explore and understand the opinions and feedback of their customers. Hence, the sentiment and opinion mining topic has emerged, which has resulted in the development of sentiment analysis tools. Due to such practical reasons, applications that combine human-based computation and machine learning techniques have become widely evident.

Hybrid Human-Machine Information Systems refer to the class of information systems that in addition to automated data processing tools such as machine learning techniques, they incorporate crowdsourcing techniques in the pipeline of *collecting*, *processing*, *storing*, as well as *disseminating* data and information. By combining human-based computation and machine learning, these systems have the potential to accomplish complex objectives by solving different data-related tasks and finally achieve results with higher accuracy and reliability than human intelligence or machine learning alone can.

4.3 Overview of Existing Hybrid Human-Machine Designs

As the main objective of hybrid approaches is solving data-related tasks with superior results, by combining the complementary strength of human cognitive skills and the efficiency of the machine learning algorithms. Designing hybrid human-machine frameworks is driven by the nature of the *task* to be solved [DCL+21], which is an important element for hybrid design decisions.

There are various *tasks* that can be solved by hybrid frameworks. For instance, *image recognition* [DDS+09] are common tasks that detect objects in images (image detection), classify the objects (image classification), and localize the objects within images (image segmentation). Beyond image processing, *classification* tasks are generic and utilized in different areas. For instance, text classification tasks involve various problems in the Natural Language Processing field such as sentiment analysis, sarcasm detection, entity recognition, etc. Another category of tasks is *prediction* or forecasting of future events, such as stock price prediction, weather prediction [NVL+14] etc.

Hybrid human-machine designs consist of two main components that interplay and complement each other into two forms [DCL+21]: i) *augmented human intelligence*, where automated data processing tools are in the loop of the human decision-making process, and vice versa, ii) *augmented machine intelligence*, where human feedback is in the loop of machine learning pipeline. Figure 4.2

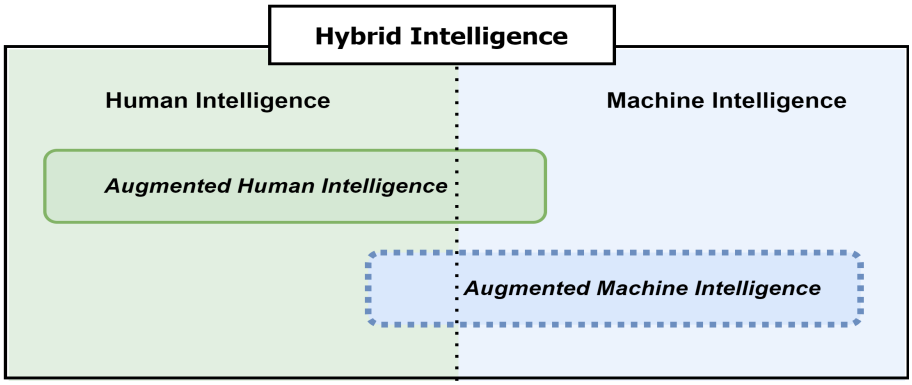


Figure 4.2 Augmented Machine and Human Intelligence

illustrates the intersection between augmented human and machine intelligence. In the following, we elaborate on the two designs.

4.3.1 Augmented Human Intelligence

Considering the rapid growth of data available on the Web and within organizations, data has become the most important factor for decision-making. However, processing large multi-source and multi-modal data requires efficient automated processing tools, which we refer to as *machine intelligence* component. Augmented human intelligence focuses on applications where human decision-making is supported by machine intelligence tools. These tools can efficiently assist by aggregating, structuring, filtering, and providing actionable data to humans which are utilized further in the decision-making process.

Machine learning tools have been shown to improve the outcome of human decision-making. For instance, in the field of the rule of law, machine learning models can assist in the decision-making process of the court, however, the final decision is made by humans [GHEG19]. Such models, if not designed and trained with balanced and high-quality data, can lead to biased results, such as racial discrimination [TL18].

A more sensitive field where fully outsourcing decision-making to machine intelligence tools is not feasible is healthcare management. However, digital health technologies can support more evidence-based decision-making in health care, and help towards personalized medicine [Hol16]. Another application of augmented human intelligence is in the financial services field, such as credit risk assessment. For instance, the data-driven models can analyse different sources of data in high volume and help in assessing the credit risk of potential loan applicants, but the final decision is taken by the credit officer.

4.3.2 Augmented Machine Intelligence

The main goal behind the augmented machine intelligence design is that machine learning algorithms can benefit from cognitive human skills. Humans provide assistance to the machine learning process to help the model train and enhance the quality of the results. There are numerous ways humans can provide input in the augmented machine intelligence setup [LM14], and here we outline two main setups: i) *machine trainer* where human feedback is used to train machine learning models, and ii) *machine improvement* where human feedback is required to enhance results from machine intelligence.

The development of machine learning models depends on data. If models are trained on inaccurate and irrelevant data, results are poor. Hence, high-quality training data is essential to successfully deploy these models. The potential of machine learning applications is very wide, covering various fields in the research and enterprise worlds. However, the fundamental problem when implementing machine learning pipelines is the lack of availability of datasets that are specific to the problems to be solved. The lack of sufficient and high-quality training data is an obstacle when it comes to building highly accurate and precise algorithms. Generating datasets for specific applications requires the involvement of humans with the domain expertise to annotate the data. In Chapter 3, we have seen that crowdsourcing can generate high-quality training data. Data labelling is a prominent scenario of augmented machine intelligence. Here, the human input acts as a *machine trainer* and is crucial for generating training data for training any learning algorithm.

Besides providing training data, human input is utilized in relevance feedback applications. Information retrieval algorithms facilitate the relevance of documents when searching for multimedia information (text, image, video, audio). These algorithms rank multimedia data to provide high-quality content to users. Relevance feedback is used to tune and enhance information retrieval algorithms with human feedback with the goal of achieving high result rankings. The human intelligence feedback here acts as *machine improvement*. Many organizations that deploy information retrieval systems utilize this approach either in an *offline* or *online* mode. In the offline mode, the search results from queries of interest are crowdsourced on online paid platforms by generating micro tasks, and crowd responses are fed into tuning the algorithms. More frequently, the online mode is used, where system users are directly asked to provide feedback on the query search result set.

4.4 Applications of Hybrid Human-Machine Systems

Due to the availability of online crowds of human individuals on crowdsourcing platforms and the ability of these platforms to effectively process data, hybrid human-machine workloads have seen wide application in various scenarios. In this section, we present a summary of various hybrid approaches that have been implemented in different application fields.

4.4.1 Hybrid Human-Machine Systems for Natural Language Processing

The combination of human-machine techniques has been applied in the area of Natural Language Processing (NLP). For instance, hybrid algorithms are used to solve problems of *entity resolution* (ER) [VBD14], identifying duplicate records in a database, records that refer to the same entity (also known as *semantic duplicates*). The scenario in this work is motivated by de-duplicating the *Facebook places* database. This database contains hundreds of millions of records about places across the world. This database is built from user check-in to places. During the check-in process, users can add places themselves, though the same places have been added already, causing duplicate entries to be inserted in the database. The hybrid human-machine approach utilizes a machine learning model that assigns probability estimates to candidate pairs how likely they are duplicates, and then optimally selects a subset of pairs and asks humans to resolve them. A similar hybrid human-machine approach has been proposed to tackle the challenging ER task [CCH+18].

Hybrid human-machine approaches have shown potential in the *Information Extraction* (IE) field. IE methods are applied in various scientific domains, including medical, biology, materials, etc. Automated extraction and aggregation of data from text sources strongly support and fasten the process of new findings. However, automatically extracting information from source text documents often is not perfectly accurate and requires humans to validate the results. As an example, in the materials science field, extracting information from text documents and generating a high-quality database of material properties is a challenging task. This is because such data is encoded into specific scientific articles that require field expertise to understand them. This has motivated the application of hybrid approaches [TCA+17], where automated IE tools initially

extract candidates that are curated by humans.

4.4.2 Hybrid Human-Machine Systems for Information Retrieval

Another field where the combination of human and machine approaches has been applied is *information retrieval* (IR). While advances in mobile technology have enabled effortless generation and storage of large multimedia data, efficient search and access to these data remains a big challenge. Crowdsourcing has shown great benefits in collecting search relevance judgments, which are required for the evaluation of information retrieval systems [LY12]. Besides generating base data for evaluation, modern retrieval systems focus on the interaction part, where user feedback is becoming a fundamental part of the system. Therefore, achieving effective information access requires that retrieval systems implement interaction between the user and the system [Cro19].

More recent work [KJR+20] implements an interactive learning approach, where user feedback in terms of positive and negative relevance judgments are fed into training the machine learning classifier, which provides new suggestions to the user. The human feedback component enhances the traditional retrieval pipelines and provides a scalable interactive learning technique.

4.4.3 Hybrid Human-Machine Systems in Digital Health

Advancements in the machine learning field have shown great results in various fields, and even if the performance of these models is reasonable, still, in some sensitive domains such as health and justice, system users refrain from completely relying on the decisions provided by these models. In such domains, hybrid approaches where machine intelligence assists humans in decision-making are preferable.

An example of a hybrid human-machine application in the health domain is phonocardiogram classification. A hybrid pipeline has shown better results compared to the machine learning classifier alone [CGM+18]. The proposed human-machine framework combines machine learning algorithms, online crowd workers, and experts to classify heart sound recordings. Whenever the classifier performs with high uncertainty, these tasks are routed to crowd participants, and if necessary, experts are involved in the final decision.

Another use case of hybrid application in the health domain is for identifying the proper medical content for analysis. Randomized Controlled Trials

(RCTs) assess the efficacy of treatments, and these trials are the most reliable source of evidence. Yet, extracting and aggregating information in structured databases from published articles about RCTs is challenging, causing inefficient reviewing when searching for relevant reports. Identifying RCT reports from a database of published articles is not trivial for machine learning algorithms. A hybrid machine learning and crowdsourcing approach has been proposed for addressing this challenge [WNSM+17]. Whenever an ML classifier is uncertain, the task is deferred to the crowd workers, which consists of contributors with three different levels of expertise: novice, expert, and resolvers. This work suggests that hybrid approaches could be applicable in biomedical data annotation and curation scenarios.

4.4.4 Hybrid Human-Machine Systems for Multimedia Analysis

Crowdsourcing is a popular method for creating labelled datasets for training machine learning models. Since modern deep learning approaches in multimedia processing require large training data, it is necessary to deploy efficient crowdsourcing techniques [DFF+19]. Considering that crowdsourcing large datasets is costly and time-consuming, combinations of machine learning and crowdsourcing are seen as a solution in the multimedia domain.

For instance, speech transcription requires that speech is converted into text in a fast and reliable manner. Automated speech recognition tools have limitations when it comes to understanding the context of the speech, as well as when the recording is not of good quality, speaker accent or jargon, etc. On the other hand, humans have the capability of understanding the context even in such non-ideal circumstances, however, the speed of transcribing is a limitation. Scribe is a hybrid system that integrates the intelligence of crowd workers with machine learning to reliably caption speech in real-time [LMN+17]. Initially, the system assigns the same audio recording to multiple crowd workers asking them to type the text, and each of them on a different segment of the audio file. The segments can overlap, which leads to overlapping transcribed text as well. An automated technique is used to merge the provided captions, which are returned to the user with a delay of fewer than 4 seconds.

Another work [GMM+15] leverages crowd workers to correct real-time mistakes done by an automated speech recognition system. Crowd users are asked to listen to the audio and read the captions at the same time. The corrections are injected back into the transcribed text by an automated component, which identifies the most likely incorrect transcribed word, and replaces it with the

corrected one.

Hybrid approaches have been applied for image analysis as well. For instance, CrowdLearn [ZZL+19] is a hybrid crowd-deep learning system for assessing damages from imagery reports after a natural disaster such earthquake, hurricane, etc. Considering the sensitivity of this application, human intelligence outperforms deep learning models in detecting false positives, which usually are because the reported images are irrelevant or in some cases come from fake sources. However, the latency of obtaining correct responses is crucial, therefore the hybrid approach proposed by CrowdLearn provides high accuracy at acceptable latency. More recent work [KSS21] on the ImageNet [DDS+09] dataset shows that combining the probabilistic output from deep learning models with output from humans can result in a lower error rate in the task of labelling large image datasets.

4.5 Challenges of Hybrid Human-Machine Systems

We have seen the advantages of hybrid human-machine information systems and their application in various fields. Combining the two components of machine learning and crowdsourcing is beneficial only if it overcomes the limitations of the individual components. In Section 2.1, we have summarized the core limitations of relying on fully automated solutions using machine learning (accuracy), as well as the main limitations (cost and latency) of the human-based computation in Section 3.2. The ultimate goal of hybrid approaches is to leverage the complementary strengths of the individual components to overcome their limitations. Combining the speed of efficient data processing with the quality of human intelligence, the goal is to generate results with higher accuracy at a low cost and acceptable latency. Therefore, the main challenge of hybrid approaches is the scalability in terms of cost and latency introduced by the human feedback, and optimization of the accuracy.

In the following part of this thesis, we will introduce the hybrid human-machine approaches that address the challenges in this field. Initially, we detail the concepts of the proposed hybrid human-machine models and then describe in detail the implementation of the hybrid architectures, and then present the evaluation results.

Part III

Hybrid Human-Machine Information Systems

5

Hybrid Human-Machine System Architectures

In this chapter, we present the proposed hybrid human-machine architectures that address the issues described in the *Problem Statement* (see Section 1.2). To recall, advances in machine learning and artificial intelligence have enabled numerous applications that handle large volumes of data at a low latency, however, frequently, fully algorithmic approaches achieve insufficient accuracy. On the other hand, human-based solutions such as crowdsourcing have the potential to process data at higher accuracy if properly managed, however, such solutions are not scalable in terms of cost and latency. Therefore, effective large data processing tasks fall within the three-dimensional challenges space: *accuracy*, *cost*, and *latency*. Here, we propose methods that address these three challenges.

The main contributions of this work are the three hybrid human-machine designs that combine human and machine intelligence with a methodology for cost and quality optimization. Additionally, our approaches emphasize the data quality control mechanisms on the crowdsourcing component, leveraging the profiles of crowd workers (location, skills, and reputation) for the task assignment and data aggregation. Furthermore, we implement and demonstrate in practice the proposed methods, by conducting experiments with real-world data following the scenarios illustrated in Section 1.3. Evaluations of our methods elaborated in Chapter 6 show that such integrative hybrid intelligence approaches achieve high-quality data while maintaining cost and latency. The proposed hybrid intelligence approaches are generic and can be seen as highly promising for many practical data processing applications.

5.1 The Benefits of Human-Machine Hybrid Intelligence

The main rationale behind hybrid human-machine approaches is that humans and computers have complementary intelligence, which can be combined and augment each other. While there are problems that machine intelligence can solve as good as humans, yet, there are many disparate easy tasks that humans and computer algorithms can do. For instance, it is relatively straightforward for a computer program to achieve the performance on a human intelligence test or games, but it is difficult or impossible to achieve human common sense [LUT+17], or tasks that require creativity, sentiment, or emotions, things that humans are proven to be superior at. However, computers compared to humans are superior at performing repetitive tasks with fast processing of large amounts and complex data. Considering the complementary abilities that humans and machines have, they have been combined and used in two main approaches: i) machine intelligence being in the loop of human intelligence, assisting humans in decision-making; or ii) human intelligence in the loop of machine intelligence where humans provide input and support in different learning stages of the machine learning models, in order to achieve higher quality output.

This thesis considers the second approach, human intelligence in the loop of machine intelligence, hereafter referred to as the human-in-the-loop (HITL) model. In this approach, the machine learning models benefit and learn from the human input, integrating the human knowledge into the learning process. The basic rationale is that humans with their cognitive abilities can complement and augment with additional information the learning and decision process.

We go beyond the classical HITL model by providing two additional hybrid intelligence models: the joint human-machine prediction model, and the high-confidence switching model. Our proposed three methods are meant to be generic and flexible depending on the given problem. As such, human intelligence plays a crucial role in achieving high-quality data, though it has received relatively insufficient attention in the traditional HITL approaches. Therefore, effective quality control mechanisms on data generated by human intelligence (hereafter crowdsourcing) play an important role. Our hybrid human-machine methods additionally integrate methods to ensure high-quality data from crowd input. Our methodology aims to optimize the trade-off between cost and quality when solving data processing problems.

5.2 Overview of Proposed Hybrid Models

Overall, many large data processing tasks that fall within the described three-dimensional trade-off space can be solved by applying one of the proposed models. However, choosing the right model requires a better understanding of the task, where the task falls in the three-dimensional trade-off space. Therefore, prior to choosing the right hybrid model, it is required to better understand the criteria with respect to cost, latency, and accuracy.

The motivation behind hybrid models is to provide higher accuracy compared to machine learning based or human based solutions individually. However, enhanced performance comes at a price: either in terms of money, time, or both. In the following, we will introduce the three proposed designs that deal with a variety of data processing and classification tasks.

- In Section 5.4, we present the *Human-in-the-loop* design and its concepts. This design complements features extracted with automated tools with features generated via human feedback. This method is preferable for scenarios where obtaining data with the highest accuracy at minimal latency is crucial. It leverages humans' intelligence and fact-checking skills to obtain essential information that is fed into the feature space of the machine learning classification pipeline. Based on Scenario 1.2, we implement the SAMS Human-in-the-loop model, and we conduct an experiment using real-world data. Details on the experiment implementation and evaluation results presented in Section 6.1 show that SAMS-HITL method has marked a positive impact on the classification accuracy of the models.

- In Section 5.5, we describe the concepts behind the *High-Confidence Switching Hybrid* design. This method utilizes an ensemble learning technique and aggregation of crowdsourcing classification results. The main part of this design is the multi-criteria decision-making model that decides whether a task has been solved by the automated component, or if it requires human input. Depending on the confidence threshold, less challenging tasks are solved by the automated component with low latency, whereas for the more challenging tasks that require deeper inspection, resolution is found on the crowdsourcing component. This method is suitable for solving problems where the three criteria of accuracy, latency, and cost are approximately of equal importance. Following the Scenario 1.3, we implement and evaluate the efficiency and efficacy of this method. Details on the experiment and evaluation results are given in Section 6.2.

– In Section 5.6, we present the concepts of the *Joint Human-Machine Prediction* design. In contrast to the previous two approaches, this method jointly combines the results generated individually by the automated and crowdsourcing components. The automated component is based on a multi-input deep learning model that combines features extracted from the images with text-based generated features from the image metadata, complemented with semantic features extracted from the text. On the crowdsourcing component, in order to improve the accuracy of obtained results, the emphasis is put on aggregation methods that utilize the crowd worker's profile and reliability scores. Following the Scenario 1.4, we conduct an experiment where we implement and evaluate the joint human-machine model, detailed in Section 5.6. In contrast to previous methods, this method is applicable in scenarios where latency is not a critical factor, but optimizing accuracy with low cost is important.

The proposed three hybrid human-machine models are capable of addressing a wide range of data processing tasks. Depending on the nature of the task, one of the models can be chosen and applied. Therefore, we propose a higher-level decision component that as input requires evaluation of each criterion: accuracy, latency, and cost, with respect to importance. The importance score ranges between 0 and 1. For each of the criteria, the meaning is as follows:

– *Accuracy* - a score towards 1 means accuracy is of utmost importance and has the highest priority, whereas a tolerable level of reduced accuracy in the results is acceptable.

– *Cost* - a score close to 0 means cost is irrelevant, i.e. the job requester can allocate a budget as high as needed in order to meet the other two criteria that have higher importance. A score towards 1 means the cost is relevant and needs to be optimized.

– *Latency* - a score towards 1 means latency is crucial, therefore the completion of the task has to be done at the lowest possible time duration. A score towards 0 means that the latency is less important.

This is a trade-off decision, which is illustrated in Section 1.1. The scores of each of the criteria are dependent on each other. Assuming that a job requester seeks higher accuracy, one of the other two criteria (cost or latency) has to be sacrificed. For instance, if the requirement for completing the tasks is to keep the costs at a minimum, then the latency criteria need to be sacrificed. Whereas,

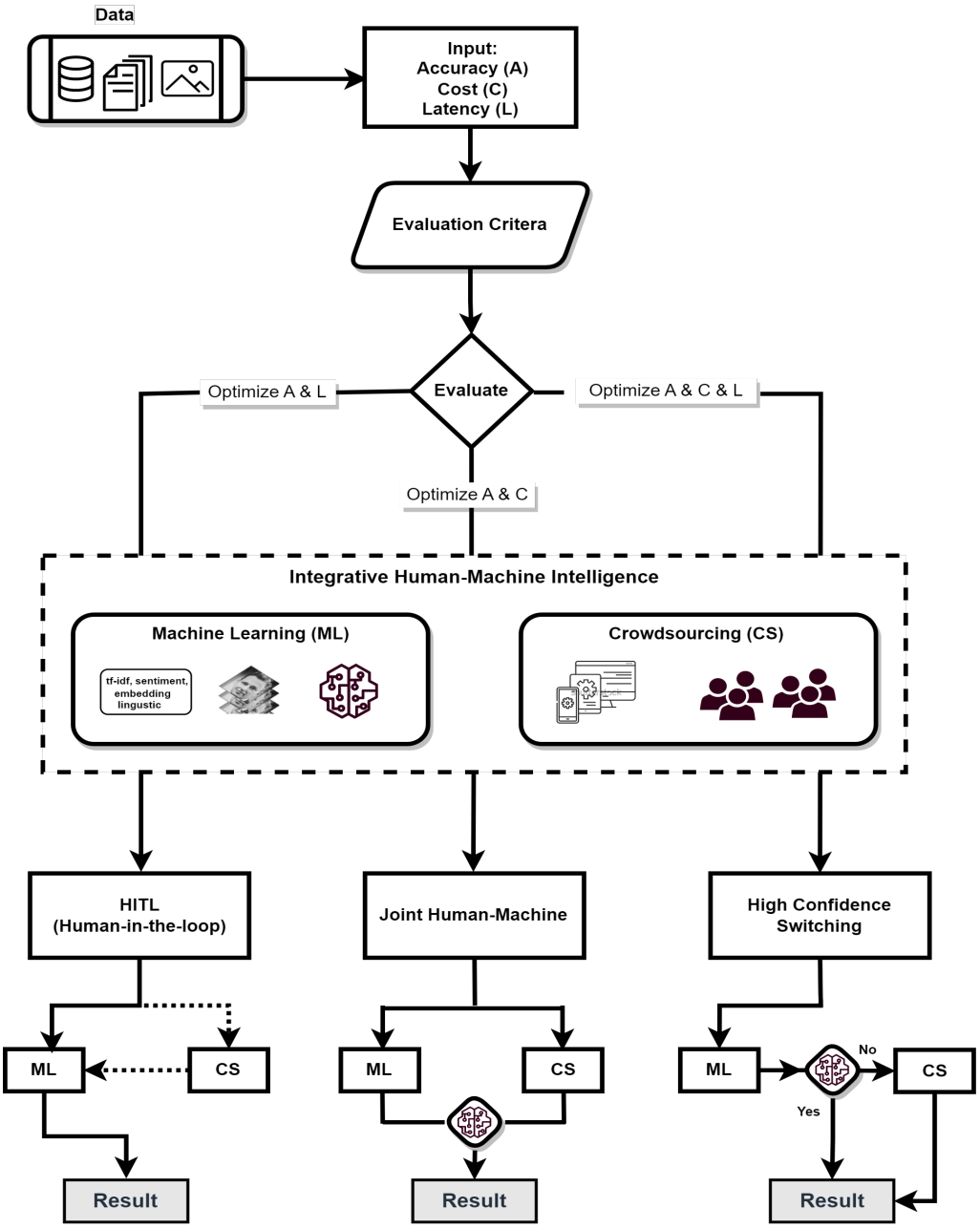


Figure 5.1 Overview of the Hybrid Human-Machine Workflow - at the start, the evaluation criteria component evaluates which of the three models is the most appropriate for solving the given data processing task. The decision is based on the weights assigned to each criterion: accuracy, latency, and cost.

tasks that require very quick resolution have to consider sufficient budget to cover the costs. A schematic overview of the selection model is shown in Figure 5.1. Initially, the requester having a list of tasks to be solved must evaluate the relative importance of the accuracy, cost, and latency criteria. Depending on these weights, the appropriate hybrid model is selected.

Guided by the schematic workflow and its description, in the following we define a multi-criteria ranking algorithm. Considering a set of models $M = \{m_1, m_2, m_3\}$, each model m_i is associated with a set of criteria $C_i = \{c_{i1}, c_{i2}, c_{i3}\}$. Based on the scenario, a set of weights $w = \{w_1, w_2, w_3\}$ is defined, where each weight w_j represents the importance of the criterion c_j. The constraint is that the sum of the weights is equal to 1. For instance, for Scenario 1, the set of weights would be defined as $w = \{0.6, 0.1, 0.3\}$ with respect to accuracy, cost, and latency. Finally, using the weighted sum of normalized criteria for each model, the highest-ranked model is the Human-in-the-Loop model.

Algorithm 5.1 Rank-Based Multi-Criteria Model Selection

Input: a set of models $M = \{m_1, m_2, ..., m_n\}$, a set of criteria $C_i = \{c_{i1}, c_{i2}, ..., c_{im}\}$, and a set of weights $w = \{w_1, w_2, ..., w_m\}$
Output: the most optimal model m^* based on the weighted sum of the ranks of its criteria.

1: **for** each model $m_i \in M$ **do**
2: **for** criterion $c_{ij} \in C_i$ **do**
3: normalize the model score

$$z_{ij} = \frac{x_{ij} - \bar{x_j}}{s_j}$$

where x_{ij} is the score of the i-th model, $\bar{x_j}$ is the mean and s_j is the standard deviation of the j-th criterion across all models.
4: **end for**
5: **end for**
6: **for** each model $m_i \in M$ **do**
7: calculate the score S_i as the weighted sum of the normalized scores z for each criterion j:

$$S_i = w_1 \cdot z_{i1} + w_2 \cdot z_{i2} + ... + w_m \cdot z_{im} = \sum_{j=1}^{m} w_j \cdot z_{ij}$$

8: **end for**
9: select the model m^* with the highest score as the most optimal model.

$$m^* = \arg\max_{m \in M} \sum_{i=1}^{n} S_{m,i}$$

10: **return** m^*

In the upcoming sections, we first outline the concepts behind the main components that constitute the hybrid models, and then we detail the three proposed models and their relationship with respect to the conducted experiments.

5.3 Hybrid Human-Machine Intelligence Concepts

The hybrid human-machine intelligence (Figure 5.2) consists of three main components: i) the *data input*, ii) the hybrid intelligence component, i.e., the *machine intelligence* and the *human intelligence*, and iii) the *data output* or results. In the following, we will elaborate on these three components in more detail.

5.3.1 Data Input

The data component is in front of the hybrid models. It can consist of data of different types and formats. For instance, input data can be multimedia records such as text, images, videos, or audio. Depending on the scenario and task to be solved, the data needs to be converted into a structured database. As an example, if the problem requires classifying text data collected from different sources (e.g., text documents, HTML webpages, etc.), the transformation part converts the unstructured data into a common structured schema (e.g., tabular format). If the problem deals with image classification, additionally the images need to be structured and metadata (path, type, etc.) about the images should be added to the structured tabular data. Once the data is properly structured, then it can be processed by the hybrid intelligence components.

As stated in Chapter 2, supervised machine learning requires labelled data in order to train the models. Typically, a portion of the data is pre-labelled. As described in Section 2.4, the supervised learning classification task can be a binary classification (two classes) problem where the dataset records are assigned to one class, or multi-class classification where there are available more than two classes, but a single class is assigned to each record. Additionally, the problem can be a multi-label classification task where there are multiple classes and each record can have one or more classes assigned.

The available labelled data is utilized to train the machine learning model, while the data input consists of numerous unlabelled data records. The ultimate goal is to annotate these unlabelled data records. Therefore, the data is organized in a way that each unlabelled data record is transformed in a task to be resolved by the hybrid component.

In scenarios where the data input has no labelled data records, labels can be obtained through the human intelligence part of the hybrid component, i.e. crowdsourcing. The amount of labelled data should be sufficient to train the model and mainly depends on the classification problem.

5.3.2 Machine Intelligence Component

The subfield of hybrid intelligence that relates to automated data processing is called machine intelligence. With this term, we refer to systems that resolve tasks in an automated and scalable manner. The goal of machine intelligence is to develop models that can resolve complex tasks, such as classification, clustering, or regression. This component is heavily based on machine learning models that are trained and capable of analysing and classifying large amounts of data.

The machine intelligence (MI) component can consist of one or multiple machine learning models that are trained with labelled data. The ultimate goal is to solve unlabelled data from the data input component. Examples of problems that can be addressed in this part can be text analysis, where pieces of text documents need to be classified, image classification where images are annotated, natural language processing (NLP) such as entity resolution, etc.

As described in Section 2.1.1, there are different learning stages when training a machine learning model: *data preparation*, *data analysis*, *feature engineering*, and *evaluation*. Part of the data preparation steps belongs to the data input component such as data collection, data structuring, and data cleaning. Whereas, the data analysis step is done here in the MI component along with the feature engineering, which is the main step of the learning stages. Depending on the nature of the task, the MI component can utilize various feature extraction techniques. For instance, if the task is text classification, then techniques such as TF-IDF or other methods described in Section 2.2.2 can be implemented to extract features from the raw input data. Extracted features are then used to train the model. If the task is dealing with image-related problems such as image classification, then image-based feature extraction techniques described in Section 2.3 can be applied.

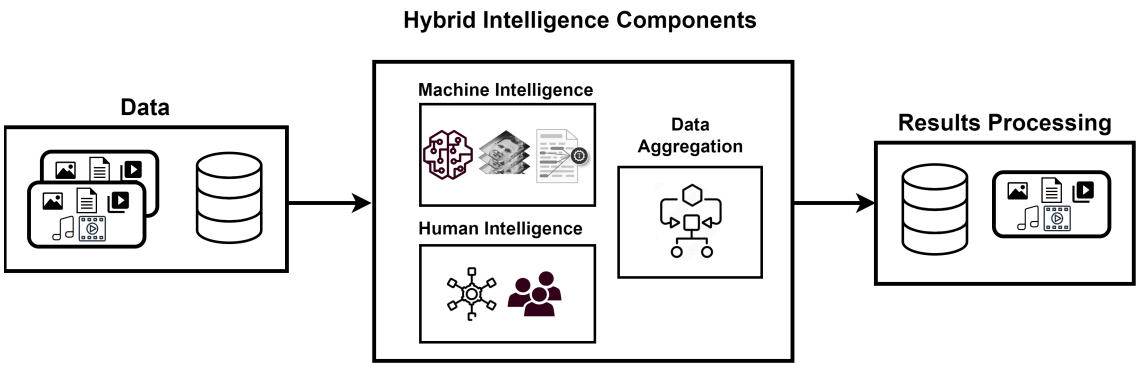


Figure 5.2 Overview of the Hybrid Human-Machine Intelligence Components

The next step is modelling, which requires choosing the right algorithm applicable to the problem. Here too, the decision is driven by the nature of the task. For example, if the problem is an image classification task, deep learning algorithms such as Convolutional Neural Networks (CNN) are more fit for purpose. In situations where the feature vector is small, which often is the case with text classification problems, algorithms such as Support Vector Machine (SVM), Random Forest (RF), or simpler models such as Logistics or Linear Regression could be chosen.

The MI component may consist of multiple machine learning algorithms as well as multiple feature-extracting techniques. In such scenarios, an ensemble learning strategy can be useful, where the decision is made by aggregating the outputs from the individual models. These voting aggregation techniques increase the certainty of the decision, thus leading to improved accuracy.

5.3.3 Human Intelligence Component

The Human Intelligence (HI) is an integral and essential component of the hybrid intelligence architectures. This subfield utilizes the cognitive capabilities of human beings. On a high level, this concept is based on the human capacity to learn, reason, and perform effective decisions based on cognitive skills. Humans possess general intelligence, which is divided into more specialized fields [Gar00] such as sports, arts, maths, logical, linguistic, musical, social, spatial, etc. Furthermore, human intelligence is harmonized into three dimensions: the *componential*, *contextual*, and *experiential*. The *componential* intelligence relates to individual established skills, whereas the *experiential* intelligence is about the ability to develop certain skills by experience. The *contextual* intelligence is about the way in which cognitive processes deal with various settings.

Besides performing tasks individually, humans can act collectively in groups, forming *collective intelligence*. The HI component here refers to the crowdsourcing concept, a method that leverages the *wisdom of crowds*. As described in Chapter 3, crowdsourcing has shown to be an effective method for solving tasks that are difficult to solve by the machine intelligence component.

The HI component consists of a large group of crowd users who are actively participating in a crowdsourcing platform. Crowd workers have different levels of expertise, skill sets, and knowledge. If not properly utilized, errors can occur that lead to low data quality. On the other hand, how the crowdsourcing tasks are designed is also important. In Section 3.5, we have described the best practices to follow in order to have successful task design in crowdsourcing projects,

and these include clear instructions and examples, task decomposition, proper crowd selection, fair pricing as well as managing data quality. Managing data quality is one of the main challenges, and it requires data quality control mechanisms. In Section 3.2.1, we have presented different data quality control aspects: qualification models (Section 3.4.1), task assignment models (Section 3.4.2), as well as redundancy and aggregation models (Section 3.4.3).

Similar to MI data resolving flow, the HI component's goal is to solve tasks that are generated from the data input component. The two components can solve the same problems, however, the main factor that differentiates them is the accuracy of the output. While the MI component can process large amounts of data at low latency, the HI component can solve tasks with higher accuracy, but at a higher cost and latency. The success of achieving high-quality data from the HI, relies on how these two components (MI and HI) are combined and interact with each other. We propose and describe three different such combinations that are generic and applicable to a diverse set of data classification challenges.

5.3.4 Results Processing

Data generated by the hybrid intelligence (combination of HI and MI components) is processed in the last stage of the hybrid pipeline, which is result processing. The result processing component can be seen as somewhat the inverse of the data input component. For each record that is unlabeled, the hybrid component assigns a label that is stored along with the data record, i.e. the class (or classes). Result processing can structure the data and enrich the data collection from the data input component. This may contribute to retrain the machine learning models in the MI component. Most importantly, the data from the result of processing can be used for other data collection and exploration purposes.

5.4 Human-in-the-loop Model

The first proposed hybrid intelligence model that combines HI and MI is the Human-in-the-loop (HITL). As described above, the MI component utilizes different feature extraction techniques to generate features that are important in the classification task. Under specific circumstances, the automatically extracted features and modelling can't achieve sufficient accuracy. However, for such tasks, humans can contribute with their cognitive and fact-checking capabilities an-

swering very specific contextual-related questions. Answers or the feedback obtained from human intelligence is jointly integrated into the MI decision-making pipeline. more specifically, this feedback is transformed into features by complementing the feature vector extracted in the MI component. The HI derived features have higher importance during the classification process performed by the MI algorithms.

This HITL model is illustrated in Figure 5.3. This model is valuable for applications where the accuracy of data is of utmost importance at low latency, and the cost is non-essential. Following the concepts of this model, we have implemented the SAMS-HITL model described in Section 6.1. The evaluations show that HITL model has a marked impact on the data classification accuracy of the models [SCS+21].

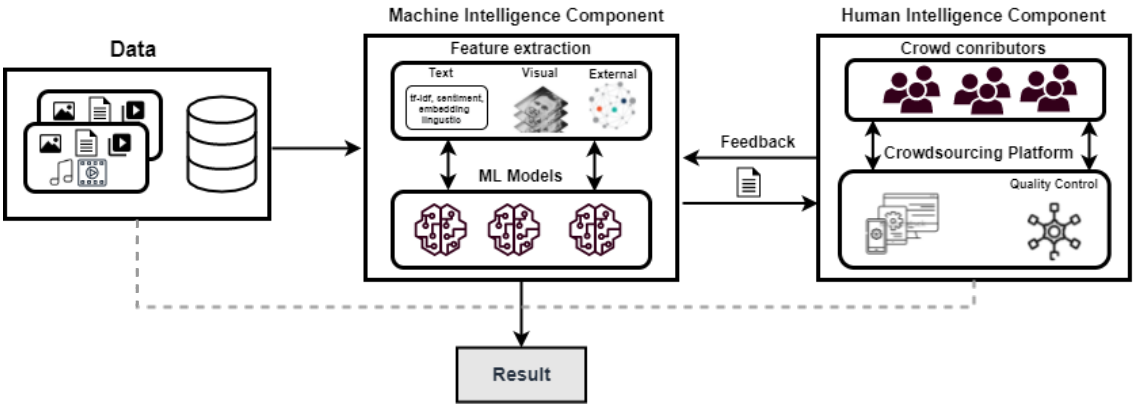


Figure 5.3 Human-in-the-Loop Hybrid Model

5.5 Confidence-based Hybrid Machine-Human Model

The second proposed hybrid intelligence model is the High-Confidence Switching Model (HCSM). While HITL model jointly integrates HI and MI features in the decision-making process, the HCSM model initially relies on the decision made by the MI component. The inclusion of the HI component in the classification task is determined based on the confidence level of the MI component in delivering accurate output. For each task generated from the data input, the result is either obtained from the MI or HI component, but not jointly.

The main idea behind the HCSM design is the multi-criteria decision-making algorithm that decides whether a task can be resolved by the MI component or

by the HI component. This is very dependent on the size and characteristics of the labelled dataset, as well as the applied feature extraction techniques and modelling on the MI component. Some classification tasks could be trivial for the MI component. This is the case with data records that have been observed previously during the training phase of the machine learning algorithms. However, there are often samples that have very different characteristics from the observed/trained records, and these have a higher likelihood of being misclassified. An advantage of machine learning algorithms is that they can estimate the confidence of their decision. These algorithms assign a probability score to each of the possible classes from the dataset. Therefore, in the HCSM model, we embed a multi-criteria decision-making algorithm that finally decides which task is required to be solved by the HI component in order to achieve results with high accuracy.

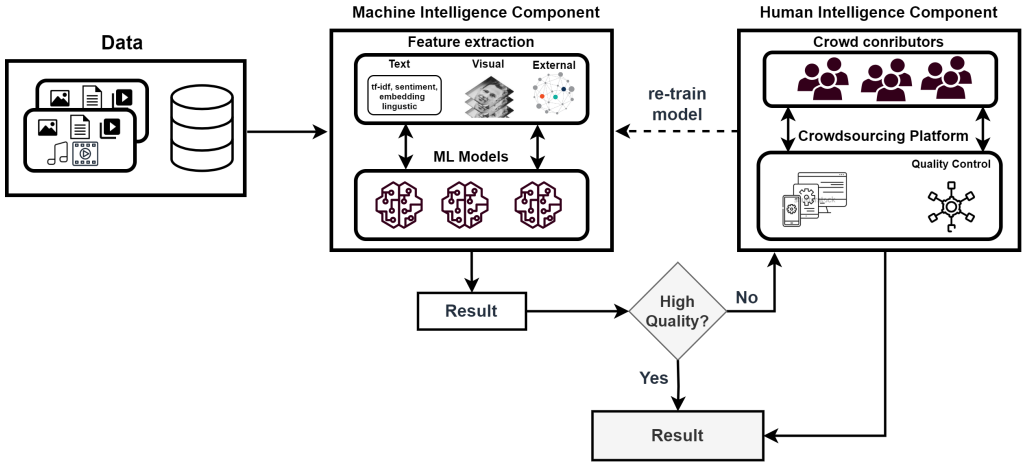


Figure 5.4 High-Confidence Switching Hybrid Model

Besides, the confidence score obtained by the MI component models, we utilize ensemble learning and voting strategy. The combination of multi-criteria (confidence and voting) is done with the ultimate goal of having higher confidence in the output generated by the MI component. The stricter we are in the criteria, the higher the chances that the MI component fails in fulfilling the applied criteria. As a result, most of the tasks are forwarded and solved by the HI component. While the HI component can provide highly accurate data following practices described in Section 3.2.1, this will lead to a higher cost and latency. Therefore, the choice of the criteria in the decision-making algorithm is a trade-off between high accuracy, low cost, and latency. The HCSM is illustrated in Figure 5.4. This method is suitable for solving problems where the three criteria of accuracy, latency, and cost are approximately of equal importance. Following

the concepts of the HCSM model, we implement and evaluate the efficacy and efficiency of this method. Details on the experiment and evaluation results are given in Section 6.2.

5.6 Hybrid Human-Machine Joint Prediction Model

The third proposed hybrid intelligence model is the Hybrid Joint Prediction Model (HJPM). In contrast to the previously presented models, here we highlight the distinctiveness of the JPM model. We have seen that the HCSM model decides to process either the result delivered from the MI or HI component, depending on the confidence of the results generated by MI component. In the HITL model, the features generated by the MI and HI components are jointly integrated, and the decision is left to the MI component (i.e. machine learning algorithms). Whereas, in the HJPM model, the *classification results* from the MI and HI components are jointly combined, and the final result is forwarded to the result processing component.

The HJPM model identifies the weaknesses of both components and aggregates these two in a manner that they will cancel out the individual weaknesses upon final output aggregation. Considering that for each task the HI component is involved, this leads to increased cost and latency. In practice, all tasks have to be evaluated by the crowdsourcing service, where multiple crowd workers are asked to annotate the dataset records.

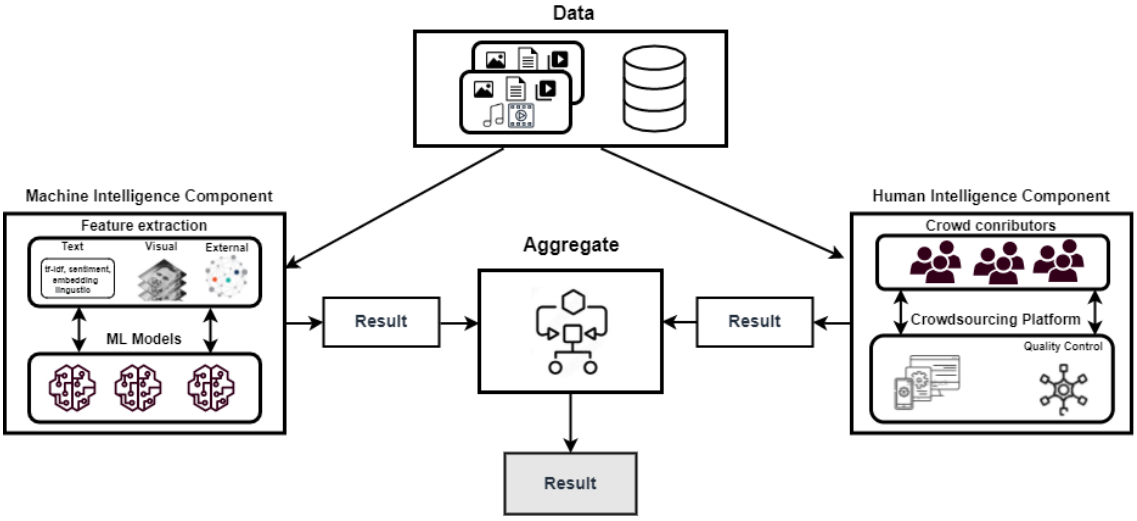


Figure 5.5 Hybrid Human-Machine Joint Prediction Model

The main idea behind the HJPM model is the *output aggregation* component within the hybrid intelligence architecture. This algorithm derives final results

by assigning weights or scores to the individual results provided by MI and HI. Figure 5.5 depicts the components of the HJPM model and illustrates their interactions. This model is suitable for solving problems when accuracy criteria is important, but the latency and cost are not. Following the concepts here, we implement and evaluate the HJPM approach in Section 6.3.

6

Experiments and Evaluations

In this chapter, we detail the conducted experiments which implement the hybrid human-machine models proposed in Chapter 5, applied and evaluated on scenarios with real-world datasets. These models address the issues described in the *Problem Statement* (Section 1.2) and are motivated from the presented *Scenarios* (Section 1.3). For each experiment, initially the motivation is introduced, followed by a review of related work. Next, the implementation is detailed, and evaluation results are presented. Lastly, a summary of the findings is presented.

In the first experiment (Section 6.1), we describe the implementation of the human-in-the-loop (HITL) model based on the proposed model in Section 5.4. Motivated by the Scenario 1 (Section 1.3), this model leverages the human cognitive skills to extract important features that are injected in the feature space of the machine learning pipeline to detect digital misinformation in social media.

In the second experiment (Section 6.2), we detail the implementation and evaluate the high-confidence switching model (HCSM) following the concepts of the proposed model in Section 5.5. This experiment is motivated from Scenario 2 (Section 1.3) to detect false news in the political domain. It shows that the HCSM is an acceptable trade-off between accuracy, cost, and latency, for scenarios where all these three criteria are equally important.

In the third experiment (Section 6.3), we present the implementation of the hybrid human-machine joint prediction model (HJPM) based on the concepts of the proposed in Section 5.6. This model joins the predictions of a deep learning component and crowd judgements. This experiment is motivated from Scenario 3 (Section 1.3), categorizing images in a cultural heritage application. The implemented model is integrated in the *City-Stories* system [SSR+17; RSS+21], which is a platform for storing, exploring, and maintaining historical collections. Additionally, the concepts of the *City-Stories* system are detailed as well.

6.1 SAMS: Human-in-the-Loop Model to Combat the Sharing of Digital Misinformation

Spread of online misinformation is a ubiquitous problem, especially in the context of social media. In addition to the impact on global health caused by the current COVID-19 pandemic, the spread of related misinformation poses an additional health threat. Detecting and controlling the spread of misinformation using algorithmic methods is a challenging task. Relying on human fact-checking experts is the most reliable approach, however, it does not scale with the volume and speed with which digital misinformation is being produced and disseminated. In this experiment, we present the SAMS Human-in-the-loop (SAMS-HITL) approach to combat the detection and the spread of digital misinformation. SAMS-HITL leverages the fact-checking skills of humans by providing feedback on news stories about the source, author, message, and spelling. The SAMS features are jointly integrated into a machine learning pipeline for detecting misinformation. The first results indicate that SAMS features have a marked impact on the classification, as it improves accuracy by up to 7.1%. The SAMS-HITL approach goes one step further than the traditional human-in-the-loop models in that it helps to raise awareness about digital misinformation by allowing users to become self fact-checkers.

6.1.1 Motivation

Advances in mobile technology have allowed for an unprecedented spread of information and both mis- and disinformation. The ease of transmission and sharing, the use of social media and messaging apps coupled with the increasing penetration of the Internet, provides a fertile ground for its spread [HPR19]. As pointed out by Ciampaglia [Cia18], the risk is the massive, uncontrolled, and often systemic spread of untrustworthy content.

6.1.1.1 Digital Misinformation Era

Digital misinformation comes in a variety of forms from entirely false to the integration of one or two misleading sentences in a piece of real news or just a provocative misleading title in the introduction to a correct piece of news. In addition to this, one also finds rumours, hoaxes, satire, and conspiracy theories contributing to what can be characterized as an online false information ecosystem [ZSB+19]. One can group such information under the umbrella term of

misinformation. More recently, one also sees a distinction between misinformation, which can be spread with or without intent to mislead, and disinformation, which intends to spread false information [HPR19].

The problem with misinformation is that it is pervasive and runs through all types of media, from print to radio to online. The latter grew considerably during the 2016 American presidential election and with the onset of the COVID-19 pandemic in March 2020 has now taken on alarming proportions. In the words of T. A. Ghebreyesus[1], director-general of the WHO, speaking of COVID-19: *"We are not just fighting an epidemic; we are fighting an infodemic"*. Digital misinformation spreads faster and more easily than this virus, and is just as dangerous. Unlike the virus, however, COVID-19 related news has two strains *true* and *false* with the latter inundating social media channels and going largely unverified. The expression *infodemic* was first used in 2003 by Rothkopf [Rot03] when writing about the SARS epidemic and highlighting the negative impact that misinformation had on controlling the then health crisis – a crisis far from the size of what we are now experiencing with COVID-19.

In a time when the world was influenced by COVID-19, we were indeed dealing with an infodemic and the question is how best to control it and fight the spread of misinformation. In order to counter this, and in light of the high level of user distrust towards online fact-checking services, there is an urgent need to train individuals to evaluate the veracity of the information that was received and shared and give them the tools to become fact-checkers in their own right.

6.1.1.2 Checking the facts

In recent years, fact-checking services have become the norm for journalists and are now also easily accessible to the public. Although the majority address issues in the political arena, SNOPES[2], a well-known service, started out primarily by debunking urban legends. Such services certainly have a role to play in response to the challenge of online misinformation [Har14; BFD18], however, there is increasing interest in coming up with an automated and scalable response [Gra18]. Despite the availability of fact-checking services, research suggests that there is a high level of distrust for such services [BF17; Hop]. Yet another argument in support of the development of an individual user application.

This research focuses specifically on digital misinformation. As this is technology-

[1] `https://www.who.int/dg/speeches/detail/munich-security-conference`
[2] `https://www.snopes.com/fact-check/`

based, the solutions considered are also often only technology-related [Gra18]. Some advocate that the response to the online spread of misinformation is through technology [KPB+20] and the integration of Artificial Intelligence (AI), others lean more towards human fact-checkers. An alternate possibility is through a combination of the two, allowing for the development of a high-level performing model used by individuals. Research in the area of mixed-initiative fact-checking [NKK+18; DMS20; Reh17] suggests that AI alone cannot be as accurate as when the human element is integrated in the fact-checking process. Human-in-the-loop AI (HAI) systems face different challenges in terms of effective performance due to the fact that individuals are involved. It is, however, possible to train models to a significant level of accuracy.

We suggest going one step further in the battle to slow-the-flow by taking an HAI approach, as well as involving those who are at the source by getting them on board as fact-checkers in their own rights. In order to do this, we developed a user-friendly tool that both identifies the veracity of the news and calls on the user to self-check four critical indicators on their own. Our proposed framework is called SAMS. Following a review of the published research and literature on credibility indicators [ERC+18; ZRM+18; LBB+18] and fact-checking guides [Bla04; PEOOAP21; BMS+18] the choice of a limited number of checks seemed to be appropriate.

In order to be best armed to counter the spread of misinformation, it is important to see who is spreading it. Spring [Spr20] suggests grouping the spreaders into seven categories ranging from the "Joker" to the "Politician" and including "well-intentioned family members". Zannettou et al. [ZSB+19] go one step further, including even bots for a total of ten categories. Keeping in mind the fact that we were looking for a limited number of indicators and yet ones that could be applied to all categories, the ones that repeatedly came to the top of the list were: *source*, *author*, *message*, and *spelling* (SAMS). We went with these and launched into the development of a prototype.

6.1.2 SAMS-HITL Architecture

Considering the level of sensitivity and criticality of the COVID-19 topic, combating the digital misinformation efficiently and effectively requires a solution that can detect on time and with high accuracy. In this situation, accuracy and latency are critical, whereas cost is not. If we refer to our proposed hybrid models in Section 5 and the multi-criteria model selection algorithm (Algorithm 5.1), the HITL model emerges as the most suitable for this scenario, as the model

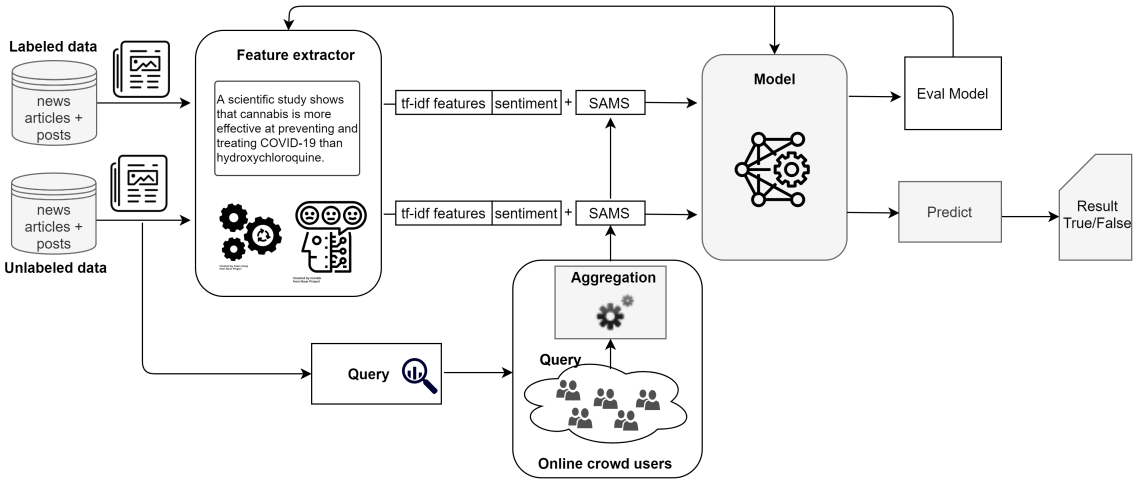


Figure 6.1 SAMS overall architecture – the feature extractor module initially performs a text cleaning step and generates the tf-idf features together with sentiment features. The labelled data includes new articles that are categorized as false or true. Retrieving SAMS features is done by querying the crowdsourcing module and collecting answers from multiple users. Judgments from users are aggregated and injected into the set of automatically extracted features (tf-idf and sentiment) and finally, the combined feature set is used by the model for prediction.

with the highest ranked score (weighted sum of normalized criteria).

In this section, we present the implemented components of the SAMS-HITL model proposed in Section 5.4. Figure 6.1 illustrates the overall architecture of SAMS.

6.1.2.1 Machine Learning Component

Considering the rapid growth of online data and spread of misinformation, and the high impact it has on society, efficient and effective data processing tools are essential. Approaches based on machine learning and deep learning techniques [SSW+17] have been comprehensively considered for fake news detection. The core component of SAMS is the supervised machine learning model which analyses the news content. This model consists of two phases: i) *feature extraction*, and ii) *model construction*.

Feature extraction is performed on the text coming from the headline and the body text. The headline of a news item is a short text that is meant to catch the attention of the reader, whereas the body text is the main part that details the news story. We consider two types of features: *statistical* and *sentiment* features based on linguistic characteristics. The statistical features are extracted using the Term-Frequency Inverse-Document-Frequency (tf-idf) algorithm, which

measures the importance of words in the text document. Analysing the sentiment of the news is very important especially when taking into account that much of the misinformation being spread started out as disinformation with the intent to deceive rather than to report objectively, sometimes for political or financial gain, and in the COVID-19 pandemic situation to exploit public fear and uncertainty. Therefore, the sentiment features focus on capturing the *objectivity* aspect, the *mood*, *modality*, and *polarity* of the reported news. Additionally, the length of news stories is an important aspect, as misinformation generated on social network channels tends to be short and catchy.

Model construction builds the machine learning model to perform the classification, in order to differentiate between false and true news. For the evaluations, we selected four different state-of-the-art algorithms: Logistic Regression (LOG), Random Forest (RF), Gradient Boosting Classifier (GBC), and Support Vector Machines (SVM).

6.1.2.2 SAMS – Source, Author, Message, Spelling

Classifying news articles by relying solely on machine learning models based on news content is a challenging task. One reason is that spreaders of misinformation have advanced their writing style and the language used in the news with the aim of distorting the truth and bypassing detection by style-based models. Another very important factor is the length of the news: false stories, especially in the era of the COVID-19 pandemic, tend to be short and alert, making it difficult to automatically analyse the message conveyed by such stories. Research in the area of mixed-initiative fact-checking [NKK+18; DMS20; Reh17] suggests that machine learning alone cannot be as accurate as when the human element is integrated into the fact-checking process.

We have identified that important features when performing fact-checking are: *source*, *author*, *message*, and *spelling*. Answering the questions about the four features of SAMS automatically is non-viable. In contrast, humans have the potential to do better, as they can perform fact-checking skills, by searching for facts on trustworthy data resources. We define a process with tips and tricks to easily answer each of the questions for SAMS. ***Source*** – taking a critical look at the source, both data and metadata, is the first step. The goal is to understand if the news stories have sources and if the sources are reliable. To better evaluate if the sources are trustworthy, we take a look at where the information originated, inspect if the references are stated and if so, trace the references checking if they are correct and trustworthy.

Author – in principle, real and serious news articles always have an author. Therefore, the first step is to identify if there is an author of the news item. If so, further inspections include if the author is a journalist, their affiliation, and academic or professional credentials. Furthermore, a check for related publications by the same author can be made.

Message – the message should be clear, balanced, and unbiased. Guidelines suggest checking for unsupported or outrageous claims, if there is a push to share the information, lack of quotes, references, or contributing sources, and identifying if the headlines provoke strong emotions.

Spelling – reputable sources will proofread material prior to publishing. Misinformation tends to have grammar mistakes such as repeated spelling mistakes, poor grammar, incorrect punctuation, use of different fonts, the writing of entire words or phrases written in capital letters, etc.

6.1.2.3 Human-in-the-Loop Approach

Obtaining reliable information related to the four aspects of SAMS is crucial for our approach to misinformation detection. While experts such as journalists are well-trained to search for the right data sources to find facts, employing them becomes expensive. Considering the amount and velocity of potentially disseminated digital misinformation, this is not scalable in terms of time.

On the other hand, crowdsourcing has been widely deployed for small tasks as it leverages the collective human skills of extensive online crowdworker communities. In specific scenarios, crowdsourcing has shown to be an alternative service to replace experts with specific domain knowledge for labelling. In this work, we design a crowdsourcing component that aggregates the inputs from multiple users to infer the true answers related to SAMS questions. The output of the crowd answers is a vector of four binary values, each value corresponding to the SAMS questions. As could be seen in Figure 6.1, the output is encoded into a binary feature vector, which later is appended to the feature vector generated using the tf-idf algorithm and the sentiment features described in Section 6.1.2.1. Finally, the concatenated feature vector is used for training and evaluating the machine learning models.

6.1.3 Overview of Experiment Dataset

In this section, after presenting the dataset we used to evaluate our approach, we describe the SAMS pipeline.

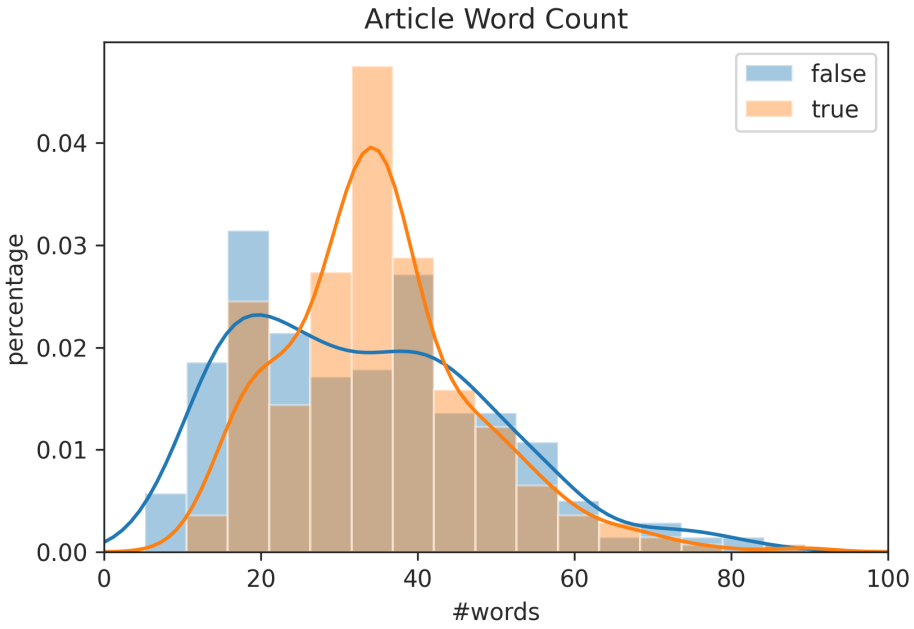


Figure 6.2 Distribution of news length by word count

In this experimental setup, we use the CoAID dataset collected and annotated by Cui and Lee [CL20]. The dataset consists of *true* and *false* news about COVID-19 from diverse sources, mainly covering websites and social network platforms. There are several types of entries such as "news articles" collected from fact-checking reliable sources, "claims" posted by official channels of WHO, "user engagement" which include Twitter posts and replies, and other "social platform posts" such as Facebook, Instagram, etc. For each entry, there is the title, abstract, content, keywords, and URL of the article or the post.

Our interest is in analysing news posts that contain potentially longer text and are posted on various social media channels (online newspapers, blogs, communication apps, etc.), therefore we focus only news articles, skipping the Twitter posts with a short text. As a result, we filtered the false and true news from the CoAID dataset, ending up with 1,127 *true* samples, and 266 *false* samples. Considering that the two classes are imbalanced, finally from the dataset we selected all 266 available false entries and 264 randomly sampled real news articles. Figure 6.2 illustrates the distribution of news length by word count, and

Table 6.1 Descriptive statistics of the dataset articles word length

Class	Min	Max	Mean	StDev	Med	Total
false	7	85	33.27	16.05	31	8'850
true	11	89	35.17	12.32	34	9'285

Table 6.1 describes the descriptive statistics of the articles' length. The average length of true and false news articles is 35 and 33 words, respectively.

6.1.4 Pipeline

The first step towards implementing a model is data pre-processing. Since this is a text classification task, text cleaning is a useful process, and that includes removing frequent words that provide non-unique information to the model (stop words) and special characters and applying stemming and lemmatization. This part is important for extracting features using the tf-idf technique. On the other hand, extracting sentiment features is possible on the raw text and this is done with the pattern.en tool [DSD12]. The sentiment features include: i) *polarity* which is given as a value between -1.0 (completely negative) and +1.0 (completely positive); ii) *subjectivity* which is a value between 0.0 (very objective) and 1.0 (very subjective); iii) *modality* feature represents the degree of certainty as a value between -1.0 and +1.0, where values higher than +0.5 represent facts; iv) *mood* feature is a categorical value based on auxiliary verbs and the answer can be either "indicative", "imperative", "conditional", or "subjunctive". Additionally, we add the word count to the feature vector.

The aim of this work is to evaluate the SAMS-HITL approach and the importance of the four indicators in the classification task. Doing that, called for having answers to the four SAMS questions for every news record from the dataset. Initially, we labelled manually the 530 dataset entries. A trained annotator used an in-house developed web annotation interface to answer the SAMS questions for each dataset entry. The annotation was a tedious task that took approximately 30 hours. After that, we designed a crowdsourcing job on the Microworkers[3] crowdsourcing platform. Each news story was used to generate a Human Intelligence Task (HIT), asking online crowd participants to provide the answers to SAMS questions which were stated as follows:

1. *Is there a source in this news article?* – Yes/No.

2. *Is there an author in this news article?* – Yes/No.

3. *Is the message of this news article clear, unbiased, and balanced?* – Yes/No.

4. *Is the spelling correct on this news article?* – Yes/No.

A HIT contained the URL of the original news story, the headline, and the body text. Crowd workers were instructed to click on the news link, inspect

[3] `https://www.microworkers.com`

and analyse the article, always considering the four questions that they were asked to answer. We used data quality control mechanisms [ABI+13] such as redundancy where for each task we asked three workers from three different regions: USA, Europe, and Asia. Analysis of demographics and dynamics of crowd workers on crowdsourcing platforms has shown that these regions are mostly represented [DFI18] and this is the case in the Microworkers platform as well. Additionally, collecting judgments from crowd workers from different regions could be an important factor, as diversity matters when it comes to labelling quality [KKMF12]. Furthermore, task design techniques [FKT+13] are an important element for high-quality data, therefore guidelines with tips and examples were part of the instructions in the crowdsourced job.

The choice of obtaining multiple judgments for the same question from different user demographics can increase the quality of crowdsourced data. Depending on the task complexity, various aggregation techniques can be applied, such as voting strategies, and profile-based or iterative aggregation algorithms [QVHTT+13]. The majority voting is the simplest as it is non-iterative and does not require pre-processing, aggregating each object independently by choosing the label with the highest votes. In our scenario, for a single HIT that has SAMS questions, we would get three responses for each of the four questions and the aggregation selects the answers with the highest votes. Since crowd workers can have different levels of expertise, profile-based strategies take into account information from their past contributions to build a ranking score, as well as incorporate additional information such as location, domain of interests, etc. The reputation score of crowd workers can be updated dynamically based on their performance on the existing task. Iterative algorithms are based on a sequence of computational rounds, where in each round the probabilities of possible labels for each object are computed and updated repeatedly until convergence. For the answer aggregation in our SAMS-HITL approach, we applied the Dawid and Skene algorithm [DS79], a model that is based on the Expectation-Maximization (EM) principle to model the worker's reliability with a confusion matrix.

6.1.5 Experiment Results

In this section, we outline the evaluation of the proposed SAMS-HITL approach. We used a 10-fold cross-validation for evaluating the performance of the models, and accuracy and f1 score as evaluation metrics. To analyse the evaluation of models and the importance of the three different sets of features described in Sections 6.1.2.1 and 6.1.2.2, we run the evaluation of the models for each setting

Table 6.2 Classification results (f1 score)

Features Set	Logistic	SVM	Gradient Boosting	Random Forest
TF	76.4	77.4	75.2	77.9
TFS	82.2	80.6	87.6	91.5
TFST	87.8	82.8	93.1	93
TFSC	87.1	83.7	94.7	93.6

separately. We consider the following combinations of features:

(i) tf-idf features (TF)

(ii) tf-idf + sentiment features (TFS)

(iii) tf-idf + sentiment features + SAMS trained annotator features (TFST)

(iv) tf-idf + sentiment features + SAMS crowd features (TFSC)

Table 6.2 shows the f1 score of the models under different feature setups. When considering only tf-idf (TF) features extracted from the headline and body text, the Random Forest model achieves the highest accuracy of 77.9%. Appending the sentiment features (TFS) showed to have a considerable impact on the performance of the models, lifting the accuracy to 91.5%. Finally, adding the SAMS features in both options with crowdsourcing (TFSC) and trained annotator (TFST) shows a positive impact on the models' performances. The Gradient Boosting classifier model achieves the highest accuracy of 94.7% with SAMS features obtained by aggregating the answers from the crowd. Interestingly, the TFSC approach performs slightly better than TFST in three out of the four models. It can be observed that sentiment features (TFS) have a distinct impact on the accuracy of the Random Forest model compared to TF features, improving the accuracy by 13.6%. The TFSC features increase the accuracy by another 2.1%. This directs us to further inspect the evaluation with more samples and test the models' performance with additional data sources. In such a scenario, we would expect that both SAMS (TFST and TFSC) features will make an even larger difference compared to the other settings.

Figure 6.3 shows the effect of the four combinations of features during the evaluation. We can see that sentiment features (TFS) have a significant impact on the performance compared to the tf-idf (TF) features. Furthermore, we can observe that both combinations with SAMS features overall indicate a significant difference compared to the TFS and TF approaches. Additionally, we evaluate the importance of features with a tree of forests and analysis shows that the

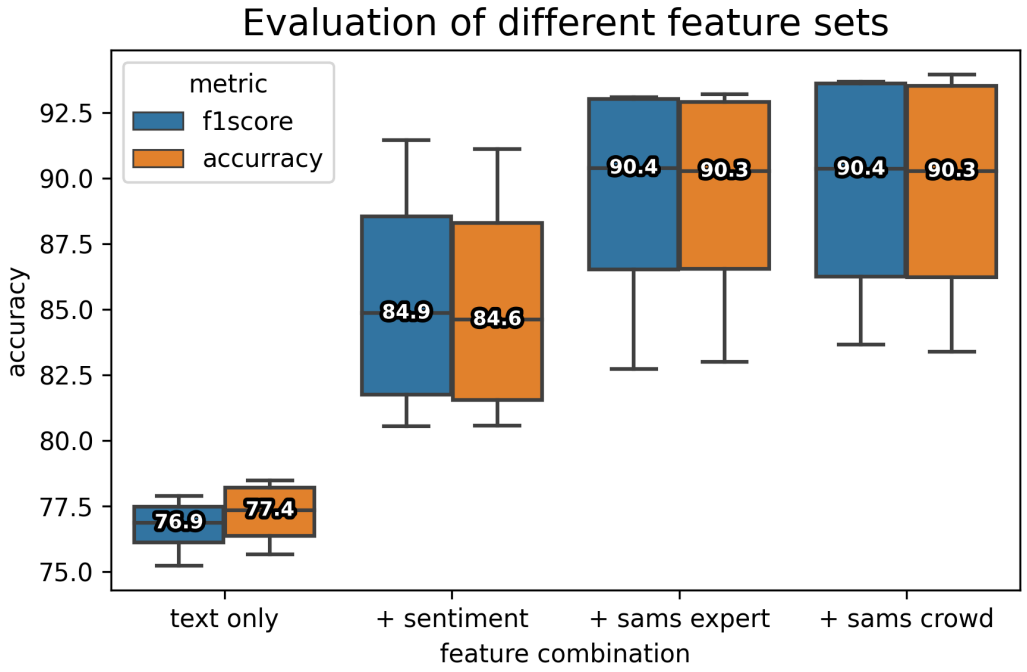


Figure 6.3 Performance impact of different set of features

most significant features are the SAMS features, where the feature about the *message* had the highest score, followed by spelling, source, and author. From the sentiment features, modality and the text length (word count) appeared in the top ten list. Finally, Pearson's correlation analysis shows that *source* feature has a moderate positive correlation of 0.57 with the class, followed by *message* at 0.51.

6.1.6 Discussion

In recent years, automatic misinformation classification has been extensively used under specific supervised scenarios. Shu et al. [SSW+17] explore the characterization of fake news in social media and the data mining perspectives. As an emerging topic, misinformation has drawn the attention of the research communities in different disciplines. As a result, several datasets have been published, related to political news [Wan17], rumour debunking [FV16], fake vs. satire detection [GMA+18], FEVER dataset for verification of textual sources [TVC+18], and a more recent dataset with news related to the COVID-19 pandemic [CL20]. Significant efforts have been made to explore the potential of deep learning [Wan17; RSL17] and the linguistic and semantic aspects [RCC+16; PRKL+18].

Crowdsourcing as a methodology can assist in the classification of news articles by fact-checking the statements. Tschiatschek et al. [TSGR+18] propose

a strategy of flagging news considered as false by social network users, and deploying an aggregation method to select a subset of those flagged news for evaluation by experts. A recent work by Roitero et al. [RSP+20] analysed how the crowd users assessed the truthfulness of false and true statements related to COVID-19 pandemic. Results show that the crowd is able to accurately classify statements and achieve a certain level of agreement with expert judgments. In response to combating the COVID-19 misinformation, Li et al. [LGS+20] developed the Jennifer chatbot which helps users to easily find information related to COVID-19. The chatbot provides reliable sources and diverse topics, maintained by a global group of volunteers.

Considering the sensitivity and the risk of misinformation spreading on one hand, and the limitations of both automated and human-based methods, hybrid human-machine approaches have been envisioned [DMS20; AHC19]. For instance, the hybrid machine-crowd [SS18] approach has demonstrated higher accuracy for the classification of fake and satiric stories. It uses a high-confidence switching method where crowd feedback is requested whenever the ensemble of machine learning models fails to achieve unanimity and high accuracy. A hybrid human-machine interactive approach [NKK+18] based on a probabilistic graphical model combines machine learning model predictions with crowd annotations for fact-checking. The follow-up user study [NKL+18] shows that predictions from automated models can help users assess claims correctly, hence tending to trust the system, even when model predictions were wrong. However, enabling interaction and having transparent model predictions has the potential to train the users to build their own skills for fact-checking.

6.1.7 Experiment Summary and Limitations

In this experiment, we have addressed the issue of classifying news stories related to the COVID-19 pandemic. We presented SAMS-HITL framework for misinformation detection. SAMS-HITL combines statistical and sentiment-based features automatically extracted from the text of the news articles with the features related to the source, message, author, and spelling of the article obtained via crowdsourcing. Preliminary results showed that the four SAMS features are the most important features of the classification model, and it has high impact on the overall classification accuracy. In summary, our proposed framework leverages the efficiency of machine learning models over a large amount of data and the quality of human intelligence for fact-checking. This method is helpful for social networks which could benefit from the high availability of their platform

members and leverage their fact-checking skills to provide feedback on SAMS questions on news articles that are posted and shared on their platform. The SAMS-HITL approach goes one step further than the traditional HAI models in that it calls on the users to answer the four questions themselves, thus raising user awareness about digital misinformation. The SAMS-HITL prototype application is currently being developed. Our objective is to help users check news articles, time train them to have a critical view, and raise awareness about misinformation. In the long run, the impact can only be positive as even without the SAMS-HITL application, people will think twice before passing news along.

A limitation in the results presented is the size of the dataset and the potential bias in the classes due to the limited diversity of sources and the length of the text in the news articles. Validating SAMS-HITL will call for its application on a much larger dataset. Having more data gives the opportunity to consider word-embedding techniques for feature extraction and the application of deep learning models for classification. Future work intends to further investigate the SAMS features. One direction is to explore the potential of automatically answering the questions related to author and spelling, which could reduce the human effort. Automated tools in combination with a customized text processing algorithm can be used to identify grammar mistakes and generate a score for the spelling. For news articles published on web portals, identifying and extracting metadata information related to the author could be done automatically. However, this is challenging for news stories published and shared via different social network channels. Further work on SAMS features will investigate the impact of using a score range instead of the binary yes/no values.

6.2 Confidence-based Hybrid Machine-Human Model for False News Detection

The rapid growth of fake news, especially in social media, has become a challenging problem that has negative social impacts on a global scale. In contrast to fake news which intends to deceive and manipulate the reader, satirical stories are designed to entertain the reader by ridiculing or criticizing a social figure. Due to its serious threats of misleading information, researchers, governments, journalists and fact-checking volunteers are working together to address the fake news issue and increase the accountability of digital media.

Automatic fake news detection systems enable the identification of deceptive news. Low accuracy remains the main drawback of these systems. The automatic detection using only news' content is a technically challenging task as the language used in these articles is made to bypass the fake news detectors. This becomes even more complicated when the task is to differentiate satirical stories from fake news. On the other side, human cognitive skills have been shown to outperform machine learning based systems when it comes to such tasks.

In this experiment, we address fake news and satire detection by implementing a hybrid machine-crowd method for the detection of potentially deceptive news. This experiment is motivated by Scenario 2 (Section 1.3) and the implementation is based on the high-confidence switching hybrid model (HCSM) presented in Section 5.5. Considering the level of sensitivity and criticality of the false news topic in the political domain, detecting false news effectively is important, as well as detecting timely and at an acceptable cost. In this situation, all three criteria (accuracy, cost, and latency) are of approximately equal importance.

If we refer to our proposed hybrid models in Section 5 and the multi-criteria model selection algorithm 5.1, the HCSM model (see Section 5.5) results as the most suitable for this scenario, as the model with the highest ranked score (weighted sum of normalized criteria). This model combines the human judgments with machine learning classification models. Whether a task needs human input or not, this decision is made based on the confidence estimates of the machine learning classification algorithms. Our approach achieves reasonably higher accuracy compared to the reported baseline results, in exchange for the cost and latency of using the crowdsourcing service.

6.2.1 Motivation

Fake news articles contain intentionally false information that could mislead the readers [AG17]. Very often this news is published with dishonest intention, designed to make the reader believe they are true in order to attain political or financial gains. The topics of fake news are very diverse, hence the problems caused by misleading stories can have a negative impact on individuals, businesses, and societies. The political deceptive news is a well-known fake news category that aims to create a biased opinion. An example of political fake news are the US presidential elections in 2016 [AG17].

Generated fake news content is hard to detect based on their content. Fake news topics cover different fields, and the used language style attempts to distort the truth. Sometimes, fake news even mention true facts but injected within incorrect context, with the goal to support a non-factual claim [FBC12].

Fake news is a longstanding problem that has affected all types of media: printed media, radio, television, and recently digital social media. The "Great Moon Hoax"[4] in 1835 is known as one of the earliest examples of fake news, in which the New York Sun published a series of articles about the supposed discovery of life on the moon.

Social media is an environment that enables the rapid production and dissemination of information at a very low cost. Due to its massive dissemination capabilities, digital and social media can reach out to millions of users within minutes. With the increase in popularity, social media has become the main source of information for many people worldwide. Despite these advantages, social media is considered to be the news production media which varies a lot from the traditional news media. Consequently, the quality of information produced by them is considered to be lower than that of traditional news media. In digital media, the boundary between news production and information creation is gradually blurring [CCR15]. Due to the low quality of news, there is a need to permanently assess the quality of news published on social media.

The ever-growing volume of fake news has turned into a significant global problem, as it is difficult to make the difference between genuine and fake news. Especially, when considering the fact that fake news has evolved, and the language used in fake news is very similar to the language used in genuine news, as the fake news are created with the intention to be trusted. Hence, fake news detection has become a very important task, yet technically very challenging. Various research studies have developed machine learning based systems to tackle

[4] http://hoaxes.org/text/display/the_great_moon_hoax_of_1835_text/

the problem of deceptive news, focusing mainly on the linguistic indicators of fake news [GKC+14; FH13; MH14; Pis17]. However, the highest achieved accuracy of detecting fake news using computer-based systems is still relatively low. Rubin et al. [RCC15] report that due to the wide variety of fake news, the usage of linguistic indicators has its limitations.

This section focuses on distinguishing satire or parody and fabricated content using the *Fake vs Satire* public dataset[5]. For this binary classification problem, initially, we apply machine learning models to automatically classify news articles as fake or satire, and features that can improve the accuracy have been identified. Later, for the same task due to the difficulty of the classification problem which requires fact-checking skills, crowdsourcing has been used as a service to get better accuracy, asking crowd-workers to classify the politically related articles as fake or satiric stories. Finally, we propose a hybrid machine-crowd approach as an advantageous solution to tackle the fake news problem in general. This approach provides higher accuracy at an acceptable cost and latency, as it combines the effectiveness of machine learning algorithms with the wisdom of crowds, through the application of crowdsourcing in the cases where machine learning algorithms fail to perform with high accuracy. This approach is generic enough and can be easily applied to other datasets and experimental setups for the fake news detection issue.

6.2.2 Overview of Experiment Dataset

In this experimental setup, we use the *Fake vs Satire* dataset collected and annotated by Golbeck et al. [GMA+18]. It is a hand-coded and annotated dataset of fake news and satirical stories, consisting of 283 fake news stories and 203 satirical stories covering only American politics.

Table 6.3 Fake news vs Satire dataset classes

class	columns	records
fake news	title, description, url	283
satiric news	title, description, url	203

All samples are extracted from online articles posted between January 2016 and October 2017 from diverse online sources. A restriction rule of allowing no more than five articles from a single website has been applied, minimizing the chance of fitting the source and language or style of writing. Each collected

[5] `https://github.com/jgolbeck/fakenews`

and annotated sample has been reviewed by two researchers, which had an inter-rater agreement given by Cohen's kappa of 0.686 and accuracy of 84.3%. Authors performed automatic classification of the news stories and initial results achieved an accuracy of 79.1%. They use a Naive Bayes Multinomial algorithm for training and testing, with 10-fold cross-validation.

Table 6.3 presents the overall statistics of the benchmark dataset. Each record has a title, body-text, and the url to the source website. We observed that 22 articles were missing the url and the reason for this can be that websites publishing fake news tend to delete the content after some time. Furthermore, nine articles with the same title have been extracted from different sources, where three of them have modified content, usually, the second source copies the content and slightly alters it.

Table 6.4 Descriptive statistics of the dataset.

#words	Mean	StDev	Min	Max	Total
article title	11.6	3.66	2	27	5,634
body description	415.2	425.6	15	5,216	201,766

Table 6.4 provides descriptive statistics. We analyse the text available in the title and the description. We can observe that titles are relatively short, with a mean of 11.6 words and a standard deviation of 3.66. The minimum length of statements is very short consisting of only 2 words (e.g., "NFL surrenders"), whereas the longest title consists of 27 words. The description part of the articles has more text with an average of 415.2 words and a high standard deviation of 425, whereas the shortest articles description has only 15 words whereas the longest contains 5,216 words. Figure 6.4 illustrates the ratio of words per article description.

Example records from the dataset are presented in Table 6.5 where the sample on top is a fake news whereas the bottom one is a satire.

6.2.3 Pipeline

Below, we describe the machine learning pipeline applied in the classification task of distinguishing between false and satiric news. The emphasis is put on the feature extraction techniques, detailing the two groups of features that were extracted solely from the text of the news stories.

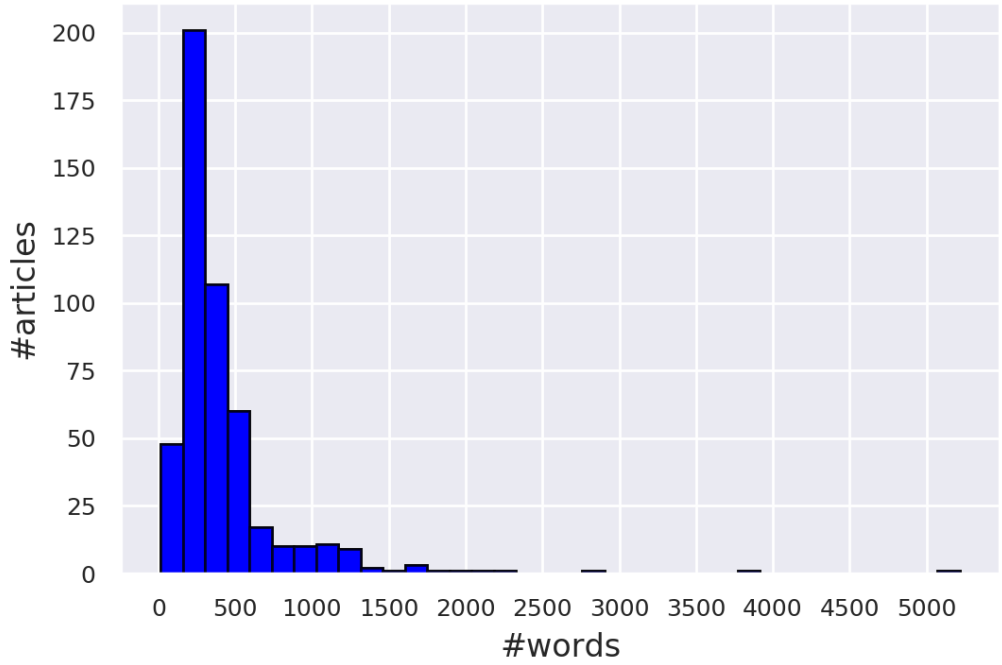


Figure 6.4 Articles body length distribution on the dataset

6.2.3.1 Dataset analysis

Applying machine learning classification methods requires that the dataset have sufficient samples, and especially analysing the news articles requires rich text content in order to apply text analysis. The dataset used in this work is not big, however, it has rich content. We analysed the corpus of words used in the title and description part of the news articles, and we found that the title together with the body content of the articles has in total of more than 207 thousand words and a vocabulary of 29,816 (distinct) words. In this work, all features are extracted only from the text content taken from the title and the description.

6.2.3.2 Feature extraction

We run an initial analysis to find features that could have significant importance to the classification problem. We identified two groups of features: (i) text-based features, (ii) features extracted by querying a search engine. *Text based* features extracted on the text taken from title + description part are listed and explained as follows:

– *tf-idf* - we compute term-frequency, inverse document-frequency, with following parameters: *min_df=3*, *max_features=3000*, *stop_words='english'*, *tok-*

Table 6.5 Two example records from the dataset: fake and satire

title	Marijuana now legal in state of Texas
body-text	In a 10- 2 vote marijuana has now been legalized in the state of Texas. Texas first marijuana dispensaries said to open up in Corpus Christi, TX April 12, 2017. We were out speaking with local residents of Corpus Christi today who were very excited to say the least of the legalization.
source	www.react365.com
classified	**fake**

title	Obama Loses Pickup Game Against Kobe, Takes Out Frustration On White House Staff
body-text	WASHINGTON – After taking a beating in a one-on-one game of basketball against Kobe Bryant today, the President shutout every member of the White House Staff in 48 consecutive games of 21. “I knew Kobe wasn’t going to let me win,” Obama said. “But after that 21-3 rattling, I made the ” staff pay, he added smiling. ... “It was quite funny to see actually,” Jackson said. “I heard all these things about the Pres. being a pretty good ball player, but Kobe took him to school.” Bryant was pretty pleased with the result, but said he expected to win.
source	www.sensationalisttimes.com
classified	**satire**

enizer=word_tokenize, *n-grams=(1,4)*, *analyzer='word'*, *use_idf=1*, *smooth_idf=1*, *sublinear_tf=1*

– *paralinguistic* - we compute paralinguistic features using the LIWC feature extractor [PFB01]. LIWC uses a dictionary that contains almost 6,400 English words. The categories represent percentages of word occurrences, and each word can fall in several categories. This results in a vector with 93 categories, including word count and punctuation.

– *sentiment* related features extracted with *pattern.en* tool [SD12]: *sentiment* feature contains the tuple (polarity, subjectivity), where polarity is a value between -1.0 and +1.0, and subjectivity has a value between 0.0 and 1.0; *modality* feature represents the degree of certainty as a value between -1.0 and +1.0, where values higher than +0.5 represent facts.

The second group of extracted features relies on querying a search engine. Our assumption is that fake news online providers do not keep their content

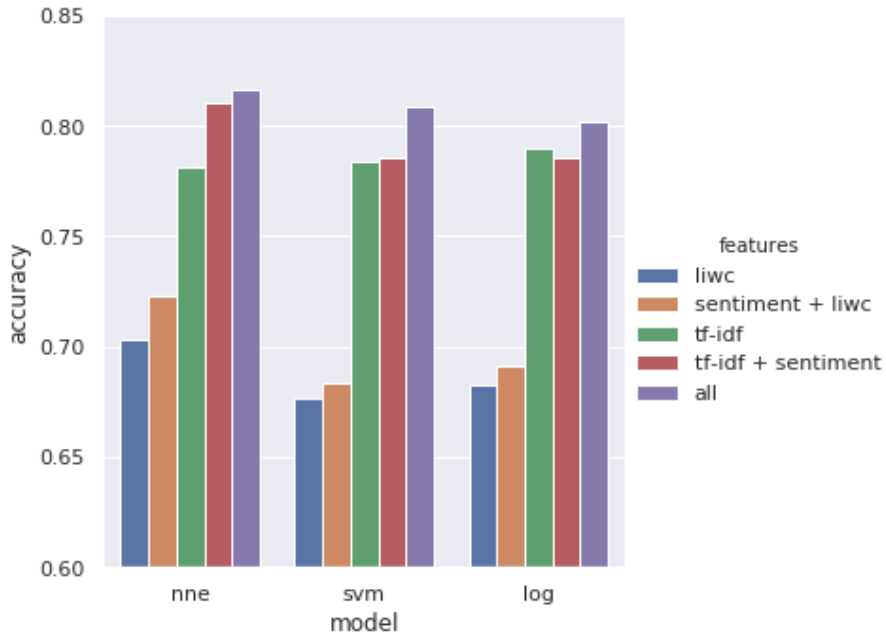


Figure 6.5 Classification accuracy of models with different set of features

available for a long time, therefore querying a search engine and checking if content exists on its database or not is very important information. We chose Google search engine to query the *title* of each sample in the dataset and analysed the first 10 entries on the result set. For each document in the result set, the title found in the HTML header and the snippet has been extracted. From this available information, 2 features based on the Jaccard similarity method have been calculated:

- the average similarity score between the dataset sample title and *titles* from the search engine result set
- the average similarity score between the dataset sample title and *snippets* from the search engine result set

6.2.4 Experiment Results

This section presents the results of the conducted experiment, where we explored three different approaches to address the detection of fake news. The first two approaches utilized distinct techniques and algorithms, i.e. machine learning and crowdsourcing, whereas the third approach is a combination of these two. In the subsequent section, we analyse the outcomes and present the results of these approaches, providing insights into their effectiveness and performance.

6.2.4.1 Classifying News using Machine Learning

For evaluation, we selected 5 different machine learning classification models: Logistic Regression (Log), SVM, Random Forest (RF), Neural Networks (NN), and Gradient Boosting Classifier (GDC). Results are presented in Table 6.7. We follow the setup and evaluation approach reported in the baseline paper [GMA+18], running a 10-fold cross-validation on the entire dataset. To implement and run the evaluation, the Scikit-learn toolkit [PVG+11] was used. The *StratifiedKFold* cross-validator with no randomness and shuffling provides constant train/test indices to split data in train/test set. It provides the same folds every time the classification is run.

To analyse the importance of the generated features, the evaluation has been done by considering the group of extracted features individually and combined. We consider the following features individually: i) LIWC features; ii) tf-idf; and the following as combinations of different features: iii) LIWC with sentiment and Google-related features; iv) tf-idf + sentiment features and; v) all available features (tf-idf + LIWC + sentiment + Google text similarity features). Figure 6.5 illustrates the accuracy of each of the 5 models considering the above-mentioned combinations of features. We observed that tf-idf features alone achieve the highest mean accuracy, for instance, Logistic Regression 78.95%, and adding the other set of features increases the accuracy. Making use of all features gave the best result, where the Neural Network model achieves overall the highest accuracy of 81.64%.

6.2.4.2 Classifying News via Crowdsourcing

Automatically classifying news articles related to politics is a challenging task. We identify two potential reasons for this. First, as the machine learning models rely on text information only, the length of the available text data and the number of samples is crucial. The dataset we evaluate is small and contains a low number of samples. Second, this is about politics, the language used by news providers, especially providers of satire is advanced and does not correspond to a classical fake news detection problem, hence classification using linguistic approaches mostly fails in this case.

In contrast to this, humans have the potential to do better in this scenario, as fact-checking skills, i.e. searching for facts over available online data and information do help in evaluating these news articles. Expert domains (journalists) are well-skilled in searching for the right data sources to find facts. However,

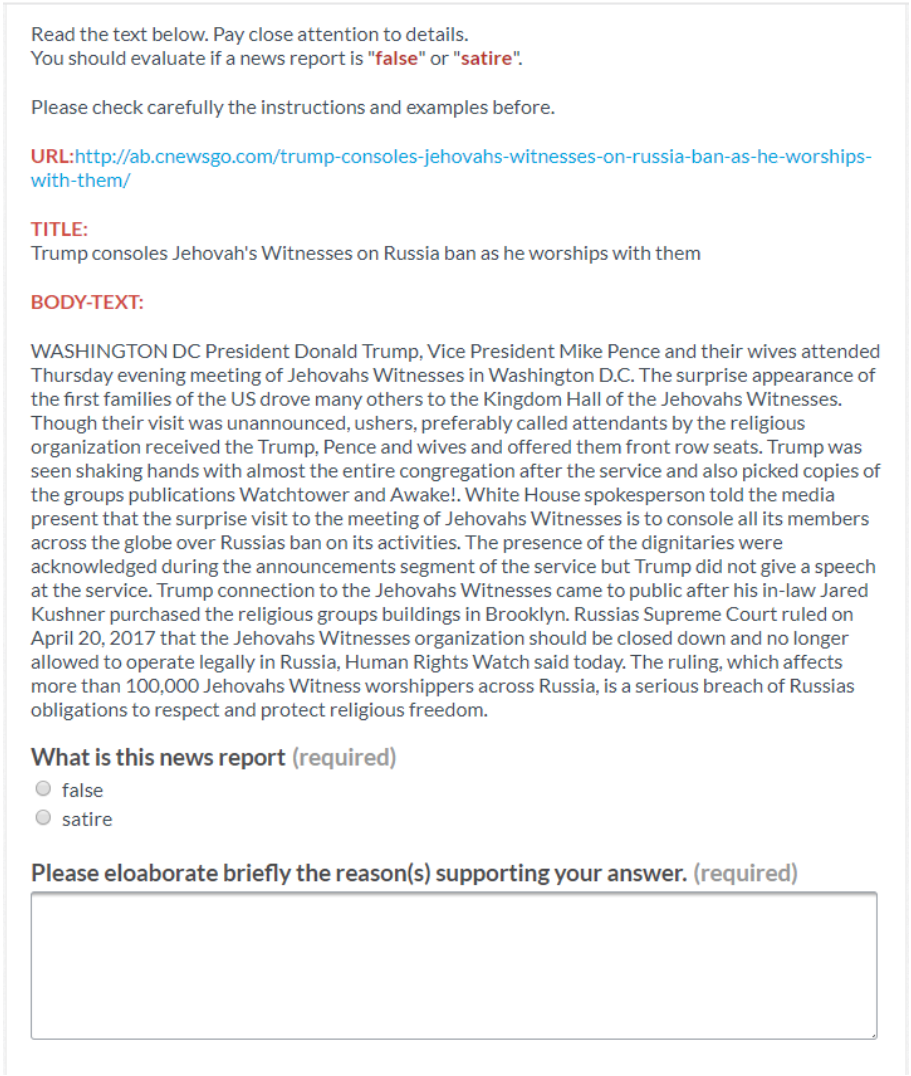


Figure 6.6 Example task designed on Figure Eight platform

employing experts becomes expensive, moreover, it becomes slow considering the amount of potentially disseminated fake news stories such that it might result ineffective in scenarios where prevention is required.

As a result, crowdsourcing has been considered as an alternative solution to this problem. We make use of Figure Eight[6] crowdsourcing platform to generate HITs (Human Intelligence Tasks), asking online crowd workers to provide a judgment for articles from the dataset. An HIT contains the title accompanied

Table 6.6 Results obtained via crowdsourcing.

dataset	samples	accuracy	cost	latency	workers
fake vs satire	486	84%	$78	29 hours	79 / 1088

[6] https://www.figure-eight.com/

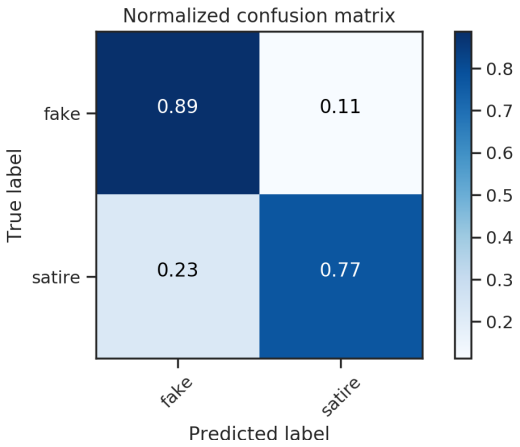


Figure 6.7 Confusion matrix - results obtained from crowdsourcing service

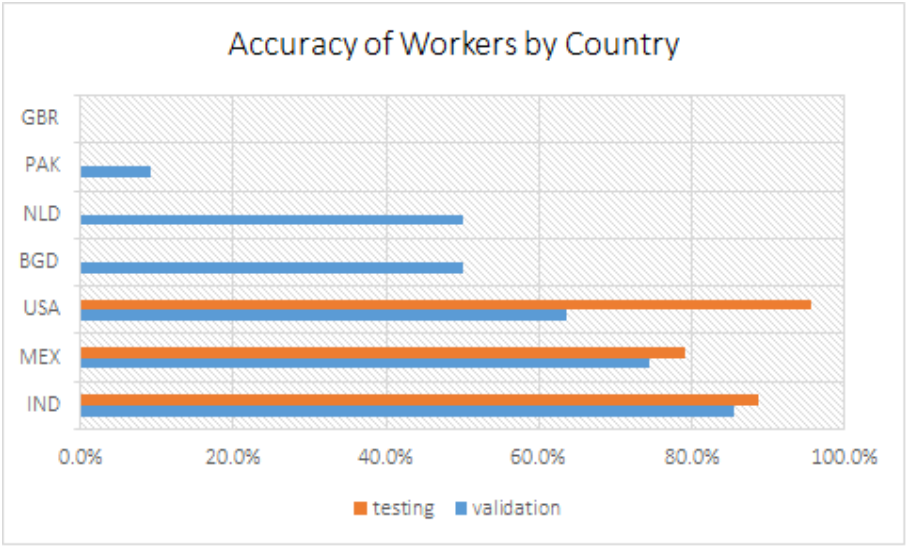


Figure 6.8 Accuracy of crowd workers based on country

by the body text of the news article, and if available, the URL address to the source content provider. Crowd workers are expected to analyse the content, if necessary, search for facts on web resources, and provide the answer which is one of the two classes: fake or satire. Figure 6.6 illustrates the designed job on the crowdsourcing platform. Task design techniques [FKT+13] are important factors that increase the quality of crowdsourced data, therefore elements such as clear instructions, rules, tips, and examples have been carefully considered making it clear and simple for the online workers to solve the tasks.

The entire dataset was taken to generate 486 HITs, one HIT per news article. In order to ensure quality control, several mechanisms have been applied. First, the golden question [OSL+11] and gold-injection methods [LLO+12] have been enabled, which disqualifies users that perform with accuracy lower than 70% on

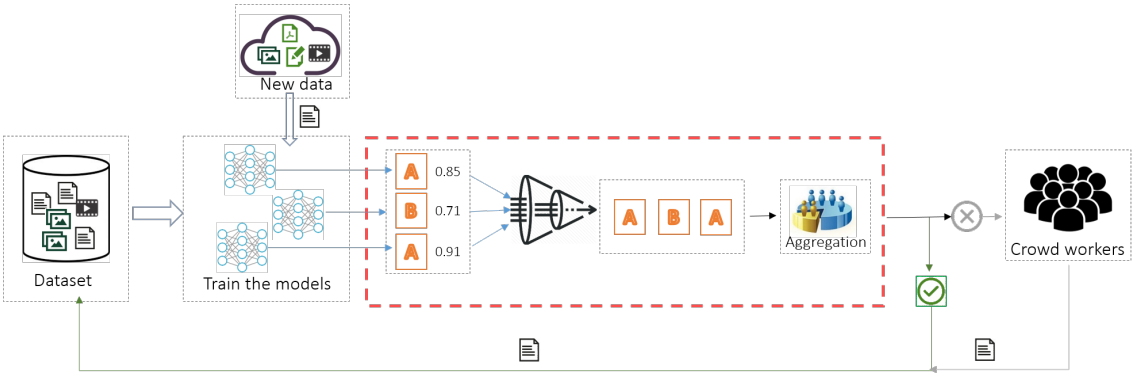


Figure 6.9 The high-confidence switching hybrid architecture for fake-news detection

the test questions. Second, the reputation mechanism [DAP15] is enabled, asking 2nd-level (middle-ranked experienced) workers and the 3rd-level workers who are most experienced and have the highest reputation. Third, the workers set was filtered by adding "English" as an extra requirement, and we limited the tasks to workers only coming from a short list of 15 countries (including the USA, Canada, UK, India, Pakistan, etc.). Last, the redundancy technique has been applied, asking 5 different workers to undertake the same task and voting mechanism that aggregates the most likely correct answer. The reward for this task was $0.02 and the entire experiment had a cost of $78. Table 6.6 shows the results obtained from the crowdsourcing experiment. Applying these quality control mechanisms led to an accuracy of 84% on the entire dataset. But, on the other hand, these mechanisms dramatically reduced the number of workers that contributed to this job to only 79. That directly affected the duration of the job (especially the redundancy approach) making it slow and completing after about 29 hours. Figure 6.8 illustrates the accuracy of workers grouped by country.

6.2.4.3 Hybrid Machine-Crowd Method

In general, automatic fake news detection with machine learning is a challenging task and the achieved accuracy is not very high due to the difficulty and nature of the classification problem. On the other side, crowdsourcing has the potential to provide higher accuracy due to the human cognitive skills required for this problem, however, it is slow and expensive. As a result, a hybrid machine-crowd approach is a trade-off solution that improves accuracy.

Our proposed hybrid approach is illustrated in Figure 6.9. On the left part, there are multiple machine learning pre-trained models with historical data and

on the right side, there is access to a crowdsourcing service with available online crowd workers ready to handle tasks. Its main component is the decision-making model which decides if the answer from the automatic classification part is good enough, or if it needs a human in the loop, i.e. forward it to the crowd workers.

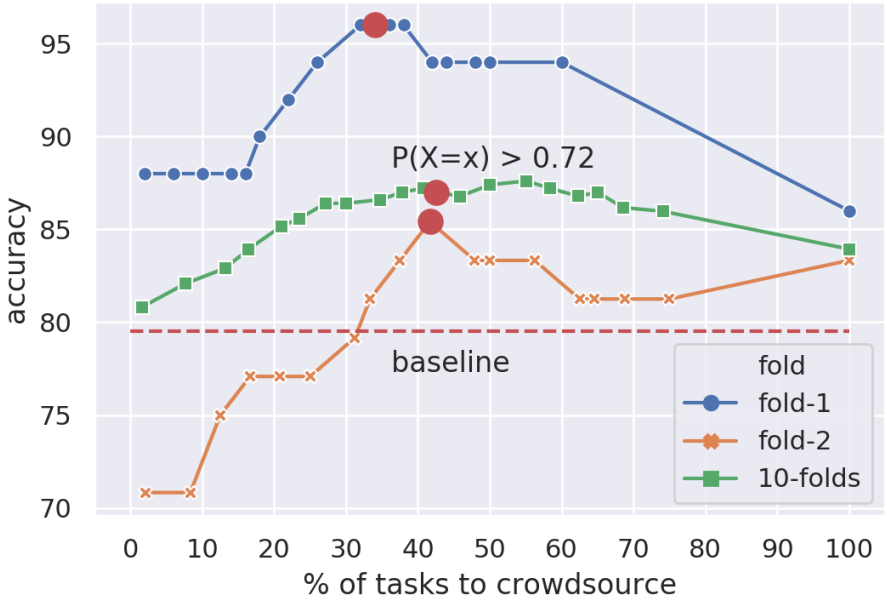


Figure 6.10 Hybrid machine-crowd approach results: Illustration of the trade-off solution between accuracy and cost & latency

The decision-making model analyses three parameters for deciding whether the automatic classification is acceptable:

- i) t - threshold on the classification probability estimates of the machine learning models, that decides which classification labels will be taken for further consideration;
- ii) v - minimum number of models that pass the threshold;
- iii) a - minimum number of models that agree on the same output label.

From the 5 machine learning models that have been initially tested, the top 3 models (Logistic, SVM, and Neural Networks) have been selected, as the other 2 models (Random Forest and Gradient Boosting Classifier) underperformed. Each model provides a probability confidence [Vov02] for the classified data record. If the three conditions are satisfied, the output from the machine learning models is considered as the final label, otherwise, this record is forwarded to

the crowd. On the crowdsourcing side, in order to choose the right worker for these tasks, as presented before, quality control methods have been applied and further analysis is done on two parameters from the workers' profiles: i) reputation score; and ii) location (country, region).

Table 6.7 Machine learning classification results.

classifier	10-fold cross-validation
Logistic	80.18
SVM	**80.82**
Random Forest	72.34
Gradient Boosting	72.13
Neural Networks	**81.64**
Baseline (Naive Bayes)	**79.1**

Analysis of finding optimal parameters used by the decision model was performed. These parameters are used as a trade-off of the hybrid approach considering the accuracy, latency, and cost factors. For comparing the performance of the developed approaches, testing, and evaluation with 10-fold cross-validation have been used in order to compare the results with i) the baseline paper; ii) our improved automatic detection approach presented in Section 6.2.3.1; and iii) results obtained with crowdsourcing service presented in Section 6.2.4.2. One split from the 10-folds has been used to test and find different sets of values for the three parameters of the decision model:
i) $T = \{0.6, ..., 0.9\}$; ii) $V = \{1, 2\}$; iii) $A = \{1, 2\}$.

A decision to run the test on the fold that provided the lowest accuracy of the three machine learning models was made. It turned out that a balanced solution that increases the accuracy up to 85% is when applying a probability threshold t=*0.72*, at least 2 models pass this threshold v=*2* and a=*2* at least 2 models agree on the same answer. Around 41% of the records are forwarded for crowdsourcing and about 59% of the records are considered on the automatic detection part.

These parameters are tested on the split that had the best accuracy on the automatic classification, and the results confirmed these parameters to be on the top performers. Finally, the test on the entire dataset running the 10-fold cross-validation is done. Detailed results are presented in Table 6.8. The results of testing the hybrid machine-crowd approach are better illustrated in Figure 6.10. Keeping the human factor in the loop and using its input on 40% of the tasks provides a trade-off solution between accuracy, latency, and cost.

Table 6.8 Results from the hybrid approach.

	fold-1	fold-2	10-folds
top *k* models	3	3	3
prob. threshold	0.72	0.72	0.72
voting	2	2	2
accuracy	96%	85%	87,2%
% of tasks to crowdsource	34	41.6	42
estimated cost	$2.7	$3.3	$34
estimated latency	1h	1.2h	29h

6.2.5 Discussion

Finding the right threshold for choosing the amount of data to be crowdsourced remains a challenge. In the automatic detection part of this work, a voting strategy among three machine learning models that satisfy a threshold for the probability estimates has been applied. This part has the potential to be improved, and future work would perform deeper analysis on modelling this part.

An important element in detecting fake news and distinguishing satirical stories from them is the source of the news. Obtaining a credibility score for the news content provider based on their past published data is not an easy task, especially when there is little information on the news content provider. In this scenario, asking online crowd workers to analyse and assess the credibility of social media news providers would be a potential solution. These scores then can be injected as an important feature into the automatic detection models. This approach of assessing the credibility of social media news providers has a drawback as it can lead to low-quality data because crowd workers might be subjective when grading the content provider (e.g., crowd workers might be biased because of political preferences). In this case, as a counter method (quality control mechanism) we would apply the strategy proposed by [SZD+17] for detecting vulnerable workers specific to this case.

6.2.6 Experiment Summary and Limitations

In this section, the detection of fake news vs satirical news using the *Fake news vs Satire* dataset has been addressed. Initially, the automatic detection using machine learning models was analysed and evaluated. The approach of using a combination of features extracted only from the text content in the news ar-

ticles (tf-idf + paralinguistic features using LIWC + sentiment-related features) and text similarity features extracted by querying Google, has shown to perform better than the baseline results by 2.54%. However, to us, this was still not a satisfactory accuracy level considering the sensitivity and importance of this classification task. Hence, applying crowdsourcing services to leverage the fact-checking and cognitive skills of online crowd workers can perform better in comparison to the machine learning method. This work shows that crowd workers alone can achieve an accuracy of up to 84%. However, applying crowdsourcing alone does not scale in terms of cost and latency. As a result, a hybrid approach has been proposed that combines machine learning and crowdsourcing for fake news detection.

The highlight of the proposed approach is the decision-making model, which consists of multiple machine learning algorithms that consider their classification probability estimates and an aggregation strategy to estimate whether the task needs human input or not. The final results of this work show that the hybrid approach provides higher accuracy compared to the automatic classification and crowdsourcing alone, up to 87%. Considering the specificity of the classification task, this approach can achieve decent results at low price and latency. Such a system could be applied to social media, which would leverage the human intelligence of online users to detect fake news and prevent its dissemination.

6.3 Hybrid Human-Machine Joint Prediction Model for Historical Data Classification

The advancement of digital technologies has helped cultural heritage organizations to digitize their data collections and improve their accessibility via online platforms. These platforms have enabled citizens to contribute to the process of digital preservation of cultural heritage by sharing documents and their knowledge. However, many historical datasets have problems due to incomplete metadata. To solve this issue, cultural heritage organizations heavily depend on domain experts.

This experiment is motivated by Scenario 3 (Section 1.3), addressing the issue of enriching the metadata in historical datasets. Here, we implement and evaluate a hybrid human-machine multi-input approach for historical data classification in the context of GLAMs, following the proposed model in Section 5.6. The classification task focuses on categorizing images with historical content. Considering that categorization and tagging of cultural heritage does not require emergent reaction, the latency criteria is not critical, however, the accuracy is important, but the cost should be kept minimal. If we refer to our proposed hybrid models in Section 5 and the multi-criteria model selection algorithm (Algorithm 5.1), the hybrid joint prediction model (HJPM) turns up as the most suitable for this scenario, as the model with the highest ranked score (weighted sum of normalized criteria).

Following the guidelines [Co02] for categorizing cultural heritage items, the goal of this categorization task is to enrich the metadata of the dataset by adding a new attribute, "cultural_interest". This is done by assigning each image to one or more of the following five categories: *place*, *object*, *person*, *event*, or *tradition* [OSS16]. For this multi-label classification task, we initially considered a hybrid multi-input transfer learning approach to automatically classify the images. This approach combines a deep learning model pre-trained on image visual features with a model that explores text features from the text available in the metadata. The results show that the hybrid model performs better than the individual models. Nevertheless, we consider these results imperfect. Therefore, for the same multi-label classification task, we ask online crowd workers from a crowdsourcing platform to also categorize the images. Finally, we implement and evaluate the hybrid human-machine approach as an advantageous solution. This approach provides higher accuracy at an acceptable cost and latency. It combines the effectiveness of deep learning algorithms with the wisdom of

crowds, where the judgments of crowd users and machine-based models are aggregated to increase the quality of annotations.

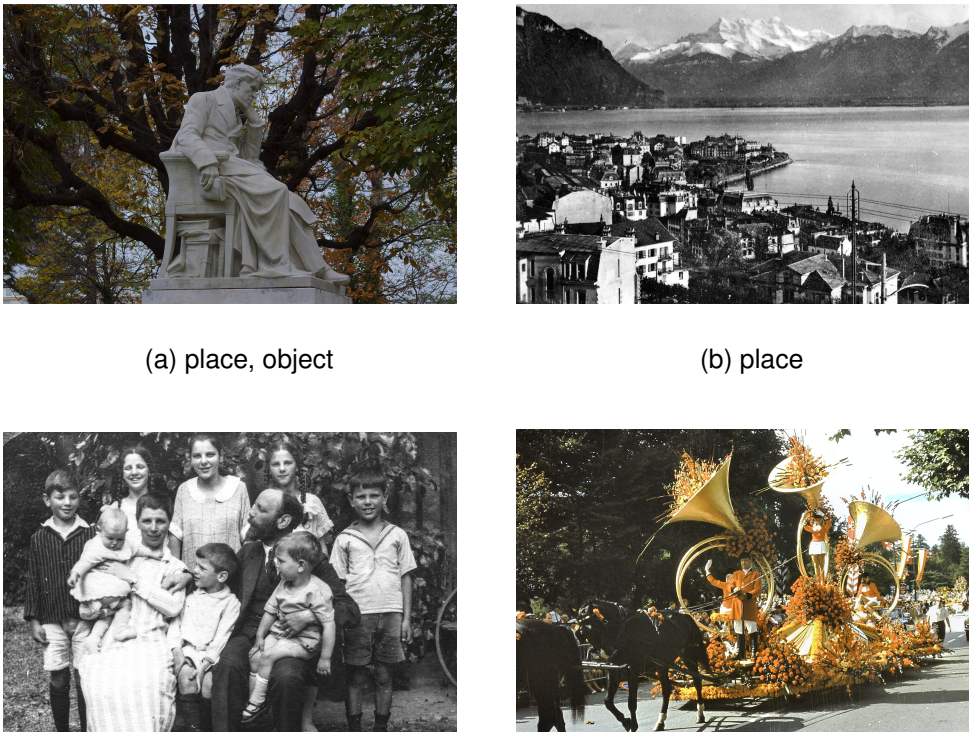
Apart from the implementation and evaluation of the hybrid joint model on a dedicated dataset, we deploy this model in the *City-Stories* platform, a system that combines entity linking, multimedia retrieval, and crowdsourcing to make historical data accessible [RSS+21]. In the following, motivation for the experiment is shown, and then in Section 6.3.3, details about *City-Stories* are provided, with a focus on the crowdsourcing component. Finally, in Section 6.2.4.3, we present the implementation and evaluation results of the hybrid model.

6.3.1 Motivation

Citizens and cultural heritage institutions have crucial roles to play in transferring historical information between generations and civilizations. Many citizens own valuable data, such as photo albums, or audio and video archives, that can be of great public interest. Considering the rapid advances in mobile internet technology, many initiatives motivate digitizing and sharing collections via online platforms. However, platforms providing reliable methods for the documentation and management of cultural and historical data only partly solve the problem, since digitized content is unusable without proper metadata. Therefore, these digital management platforms produce metadata that is of utmost importance. They cover the spatio-temporal properties of the data (date and location), the descriptive aspects of items (title, description, tags, and category), and their provenance properties (creator, owner, and licence). However, obtaining complete metadata for any item poses several challenges.

Numerous historical datasets are poorly annotated, for instance, as images may be missing tags and are not categorized. This is mainly due to annotators' mistakes or lack of accurate information. Incomplete cultural data lose their relevance in search engines and, consequently, their historical value when they cannot be found. Classifying and annotating historical data are tedious tasks, and cultural heritage institutions usually ask professionals to provide high-quality annotations. However, due to large data collections, doing in-house annotations with different specialists who have different levels of expertise in various domains is difficult and does not scale.

Considering the constraints mentioned above, many cultural heritage institutions such as Galleries, Libraries, Archives, and Museums (GLAMs) are more and more exploring the potential of crowdsourcing [OA11]. These institutions make use of the knowledge and capacity of the crowd [NOC+14], by opening



(a) place, object

(b) place

(c) person(s)

(d) even-tradition

Figure 6.11 Sample images from the NotreHistoire dataset with the corresponding annotated categories. Figures (a) and (d) have two categories, whereas Figures (b) and (c) have single categories.

their collections and by inviting online users to contribute by annotating data. For instance, the Australian Newspaper initiative from the National Library of Australia [Hol10] opened their inaccurately digitized newspapers and invited online volunteers to correct the wrongly optical character recognized (OCR) text. Another project by *Steve.Museum* [Tra09] invited online contributors to tag works of art from the museum, and tags were later compared to the museum documentation. The results showed that big proportions of tags represented terms not found in museum records. These initiatives open up collections to the online crowd to enrich data collections.

Nevertheless, the quality of crowdsourced data remains a challenge to be addressed. Compared to domain experts, who follow strict guidelines when annotating the data, crowd users are not trained and data quality is not guaranteed. Depending on the sensitivity of the data, crowdsourced annotations need an additional assessment. In some scenarios, automatic quality mechanisms such as qualification tests and aggregation mechanisms [ABI+13] would be enough to maintain a high quality of data. In other cases, it is desirable to evaluate the

manual assessment of crowdsourced data by domain experts if quality criteria suffer. However, quality control affects the latency and cost of the annotation process.

With the rapid growth of online data, having effective and efficient data processing tools is essential. Machine learning based systems have been used and comprehensively applied to solve various labelling tasks, such as image classification [DK16], object detection [GDD+15], and sentiment annotation [VC12]. Recent advances in machine learning and deep learning have increased the interest in developing tools to annotate and classify cultural heritage data as well [BBF18; LMLM+17]. However, automated tools fall short of performing as accurately as humans could. To tackle the three-dimensional problem of accuracy, latency, and cost, hybrid human-machine systems [Dem15] have been proposed. Hybrid approaches are highly promising since they leverage the scalability of machines over a large amount of data and the quality of human intelligence. Such systems are meant to combine the efficiency of computer algorithms with the wisdom of crowds [SS18].

6.3.2 The New Era of Digital Cultural Heritage

Due to the importance of preservation and diffusion, in recent years, the cultural heritage domain has demonstrated a high increase in multimedia content produced. Large cultural heritage datasets require accurate and efficient tools to organize them.

Recent works focus more on the automatic classification of historical data with deep learning techniques. These works cover image classification of different types of artworks such as paintings, statues, archaeological artefacts, and architectural object designs. Llamas et al.[LMLM+17] focus on classification with deep learning of images of architectural styles relying on visual features. Belhi et al.[BBF18] apply a multi-modal deep learning approach to predict the artists of paintings. Automatic analysis has seen application in the archaeology domain too. Work done by Cintas et al. [CLF+20] uses a dataset about Iberian ceramics, and they demonstrate the efficiency of the automatic classification of pottery vessels. In [ROFC+19] the authors combine image and semantic embeddings for the classification of statues. Classification tasks include the style, type, dimension, century, and material of the statues. Deep learning techniques have been applied to recognize characters in images of art history [MKM+19]. Their dataset consists of 2,787 images of artworks of specific iconography, and their transfer learning strategy with deep CNN models outperforms the traditional

ML techniques for their character recognition task.

Earlier work has shown great interest in crowdsourcing applications in the cultural heritage domain. Oosterman et al.[OND+14] focus on specific artwork annotation that requires domain knowledge, comparing annotations of crowd workers to experts. The task of annotating a collection of prints depicting flowers from museums showed that there is a clear relation between the difficulty of the annotation task with the performance of the crowd workers. Knowledge-intensive tasks require employing trained crowd workers. "Accurator" is a niche sourcing methodology that proposes to tailor the annotation tools to a domain and to address specific crowd communities [DDBA+17]. This methodology has been shown to collect high-quality annotations for a variety of domains. Several other works [PF14; TOH+14] focus on image tagging of artworks. The conducted experiment by Traub et al [TOH+14] to annotate oil paintings shows that introducing gamification and simplifying an expert task into a non-expert task can enable ordinary crowd workers to accomplish nearly what experts can.

6.3.3 City-Stories System

Collecting, managing, and accessing historical data is essential for the digital preservation of cultural heritage. This is particularly important for advanced applications that use digitized historical content shared across cultural heritage institutions and archives [SSR+17]. Sharing such data opens the door to several exciting possibilities.

The availability of digitized archival content shared by cultural heritage institutions raises the interest and potential for various applications that facilitate the interaction with digital archives [SS17]. First, it makes it possible to integrate heterogeneous multimedia collections from different sources, formats, and metadata schemata, to ensure access via a homogeneous interface. Additionally, descriptive metadata opens the possibility to extract the context of the documents for meaningful concepts and to link the documents across media types and external collections. Second, integrated historical multimedia content allows for interactive approaches to retrieval that support different content and context-based query types such as keyword queries, query-by-example, query-by-sketch, semantic queries, spatio-temporal queries, and any combination thereof. Third, these features allow the users of these applications to not only become content consumers, but also content providers. Citizens own valuable private collections such as photo albums, audio, and video archives. Enabling crowdsourcing as a service allows them to share important content that can be of great public in-

terest and contribute to the digital preservation of the cultural heritage of their region. Moreover, by sharing their knowledge, citizens can play a crucial role in curating existing data.

In this section, we present *City-Stories*, a hybrid system consisting of modules for multimedia retrieval, entity recognition and linking, and crowdsourcing for cultural heritage data. *City-Stories* enables the management, collection, and presentation of heterogeneous multimedia data in applications for cultural heritage, leveraging both content and metadata for multimedia documents.

6.3.3.1 *City-Stories* Concepts

In order to provide integrated access to such heterogeneous content, several important technical challenges need to be addressed:

Multimedia Retrieval The integrated content should be accessible by a very broad range of different query types, such as keyword queries to search in (manual) textual annotations, query-by-example (multimedia search with sample objects), query-by-sketch (multimedia similarity search on the basis of hand-drawn sketches), semantic queries that exploit semantic concepts and links between objects, spatio-temporal queries (i.e., queries on the location and/or time where/when a particular object has been created), and any combination of these modes.

Data Integration and Entity Linking Content coming from different sources, in different formats, and possibly also with different metadata structures have to be integrated to make sure that it can be accessed via a homogeneous interface. This includes standard approaches to schema and data integration, but also more advanced and innovative challenges like entity recognition and entity linking to make sure that links between objects (of the same or even of different media types) can be identified, stored as part of the metadata, enhanced with further external sources, and subsequently exploited for query purposes.

Crowdsourcing In addition to cultural heritage content curated by archives, user-generated content from private collections is gaining importance in touristic information systems. In order to attract the attention of potential content providers, the awareness of such touristic platforms has to be raised, the technical barrier for contribution has to be lowered, and users have to be encouraged to actively participate. This is not only true for the provision of new content,

but also for annotations to existing content (e.g., ratings or experience reports). The rapid adoption of smartphones has made it possible to also exploit mobile crowdsourcing [RZZ+15] as an efficient and easy way of reaching and using human intelligence and machine computation for solving Human Intelligence Tasks (HITs).

6.3.3.2 *City-Stories* Datasets

City-Stories data repository includes multimedia collections provided by three different historical content providers: i) *Mediatheque Valais*[7], ii) *Digital Valais*[8], and iii) *NotreHistoire*[9]. The collections mainly include images, as well as videos, audio, and documents, and span on multi-disciplines such as culture, tradition, archaeology, history, etc.

Mediatheque Valais

The first dataset was provided by the library of Canton Valais in Switzerland (Mediatheque Valais). Besides the preservation of all printed works about Valais, the cantonal's library mission is digitizing the content and making it available for exploration. The dataset contained thousands of images (~18'600), videos (~1'500), and audio (~3'300) documents with historical content of the canton, covering stories as of 1815.

NotreHistoire

The second dataset was provided by NotreHistoire, which is a participatory platform for sharing and valorizing the history and cultural heritage of different regions of Switzerland. This platform invites volunteers to publish and share their own digitized archives along with institutions (e.g., galleries, libraries, archives, museums, radio and television broadcasters, etc.). The NotreHistoire platform mainly contained images (~72'000), as well as videos (~13'400), audio (~4'600), and text articles (~1'400), and each media document has specified the license type. Therefore, for *City-Stories* a data selection process was followed, inspecting the licences of shared data and filtering only images that had a "by-nc-nd" Creative Commons license[10]. Example images from NotreHistoire dataset are shown in Figure 6.11.

[7] https://www.mediatheque.ch/
[8] https://www.valais-wallis-digital.ch/
[9] https://www.notrehistoire.ch/
[10] https://creativecommons.org/licenses/by-nc-nd/2.0/

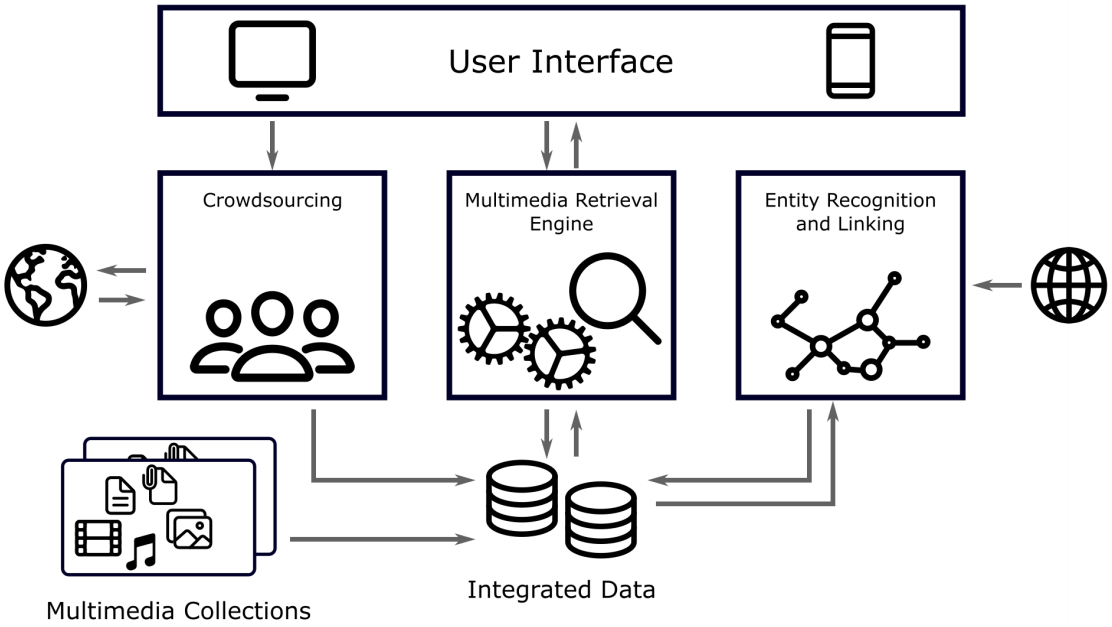


Figure 6.12 The system architecture of *City-Stories*. Multimedia collections are extracted using a common data extraction module, which is then processed separately by the multimedia retrieval engine and the entity linking engine. The module for crowdsourcing passes user-generated content directly to the two engines for processing[SSR+17; RSS+21]

DigitalValais

The third dataset contained data from the "Valais-Wallis-Digital" project, an initiative that started in 2015 and aims to digitize and share the collective memory of Valais. The dataset contained mostly images (~400), a small set of videos (~60), and articles (~10). Moreover, the dataset contained 202 images from the "Valais Mania" card game.

6.3.3.3 *City-Stories* Architecture

In this section, we describe details on the implementation of the different components of *City-Stories* and the historical collections that have been integrated. The *City-Stories* system consists of three major components:

(i) a module for *data integration and entity linking* [RHCM20]

(ii) a *multi-modal multimedia retrieval* module that is based on the *vitrivr* system [RGS14; RGT+16]

(iii) a *crowdsourcing and knowledge visualization* module [SS17; SLS18] .

During the offline phase, multimedia collections are extracted by a general data ingestion module and subsequently processed by the browser's underlying multimedia retrieval engine, while simultaneously a semantic expansion is performed. In the online phase, additional information is gained via crowdsourcing, data that is further enhanced on the fly by the entity recognition and linking module. Figure 6.12 illustrates the overall architecture of the *City-Stories*system.

6.3.3.4 Entity Recognition and Linking

The data integration component of the *City-Stories* system deals with the integration of heterogeneous multimedia collections from different sources, formats, and metadata schemata. The integration module defined a unified metadata schema and the different source's metadata schemas (e.g., XML, JSON) were transformed in order to ensure access via a homogeneous interface.

The textual information provided in the media items' metadata, specifically the title and description, were leveraged to extract and link related entities. Multimedia items in the dataset contain different information in the metadata, including text descriptions, which can be in three languages (mainly French and German, and partially in English). In the first phase, we run the *data annotation and linking*. As there are different languages in the dataset, initially we detect the language used in the text description using the language identification tool[11]. Next, we run the entity linking [MJGS+11] which automatically annotates mentions of DBpedia[12] resources in the text, and for each entity retrieved, we query the DBpedia SPARQL endpoint[13] to get additional information such as label name, abstract description, thumbnail, and image. The list of extracted entities and external knowledge base data is saved on the dataset repository, and this data is the core part of the text search and visualization.

6.3.3.5 Multi-modal Multimedia Retrieval

As initially proposed in the concept paper [SSR+17], *City-Stories* leverages a modified version of the *vitrivr* [RGT+16] content-based multimedia retrieval system, tailored to the search in historic multimedia collections. In particular, Cineast [RGS14], *vitrivr*'s retrieval engine, provides a plethora of query modes, of which *query-by-example* (QbE), *query-by-sketch* (QbS), *query-by-location* (QbL) and *query-by-time* (QbT) is enabled in *City-Stories*. QbE enables users to provide

[11] https://pypi.python.org/pypi/langdetect
[12] http://dbpedia.org
[13] https://dbpedia.org/sparql

a sample image to be looked for. Using QbS, users might sketch the query freely or modify an existing image with a superimposed sketch. Spatial (QbL) and temporal (QbT) queries allow users to search for the time and/or place where historic objects have been captured.

QbS and QbE in *City-Stories* are based on a content-based similarity search along various features. Metadata for QbL and QbT are either extracted from the multimedia objects or provided externally. Often, historical documents lack appropriate metadata like EXIF for images, and thus we heavily rely on additional data provided by the two other modules, either provided by human annotation or via semantic expansion. This data is stored in corresponding MongoDB[14] and PostgreSQL[15] databases, while the extracted multimedia retrieval features are stored in *vitrivr*'s database CottontailDB [GRH+20]. The *City-Stories* system communicates with the *vitrivr* system via RESTful API, leveraging the OpenAPI standard[16].

Figure 6.13 depicts the screenshots from the different query modes implemented and supported by the *City-Stories*query browser.

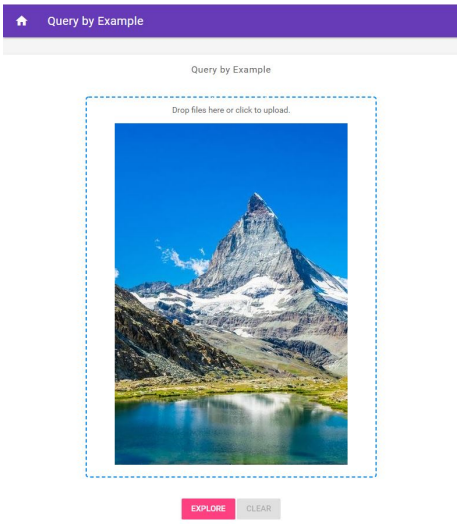
6.3.3.6 Crowdsourcing and Visualization

The crowdsourcing component of *City-Stories* allows platform users to share multimedia content. Citizens can share their private digitized historical collections and contribute to the digital preservation of the cultural heritage of their region. The cross-platform capability enables users to share their collections from desktop or mobile devices and provide metadata that covers descriptive aspects of the shared items (title, description, tags, and categories) and the spatio-temporal properties (date and location).

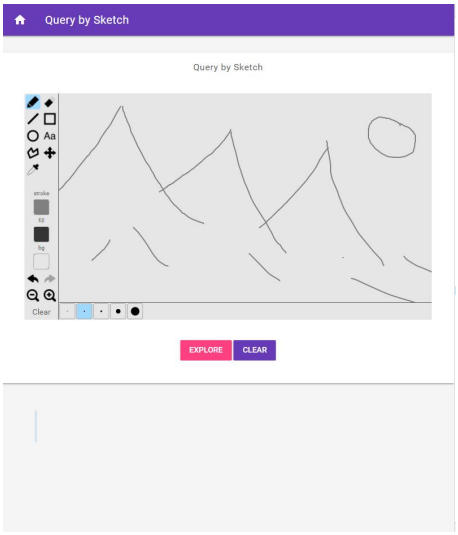
An important advantage of crowdsourcing is enhancing existing data. Historical digital collections often have incomplete metadata, for instance, images usually lack descriptive and spatio-temporal information. Classifying and annotating the data is cumbersome for cultural heritage institutions, which typically employ professionals to handle this task. Although advances in machine learning have raised the interest of the research community to develop tools to annotate cultural heritage data, accuracy remains an issue.

The type of multimedia collections considered in *City-Stories* come without or only with little geographical and temporal information. For instance, finding the

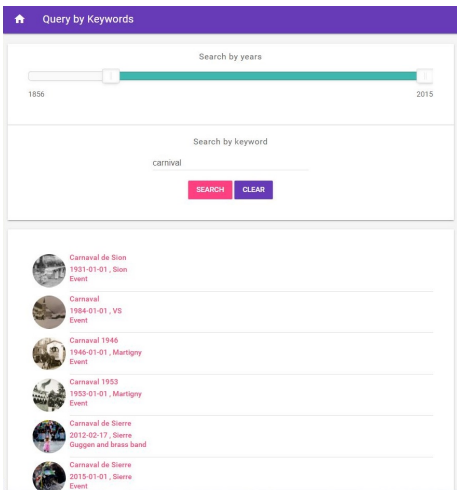
[14] `https://www.mongodb.com/`
[15] https://www.postgresql.org/
[16] `https://www.openapis.org/`



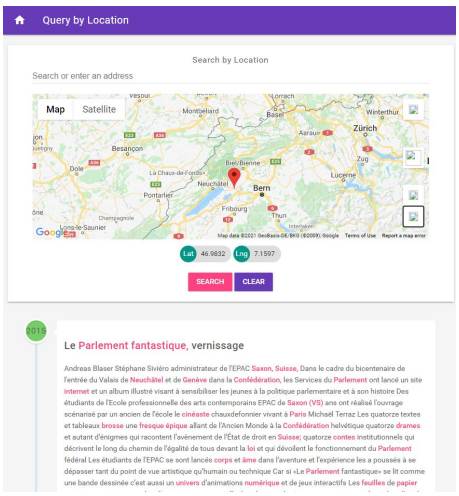
(a) Query by Example (QbE)



(b) Query by Sketch (QbS)



(c) Query by Keyword (QbK)



(d) Query by Location (QbL)

Figure 6.13 User Interfaces of the different query modes supported in *City-Stories*

location and time when and where an image was taken is a rather challenging machine learning classification task. In contrast, humans can perform better, especially if the annotation tasks are properly matched with the annotators' capabilities. We deploy four crowdsourcing tasks and leverage the wisdom of crowds to improve the metadata of historical collections:

– *Location-Finder* - users can test their knowledge about places. An image is shown on the UI and the user is asked to find the location on the map.

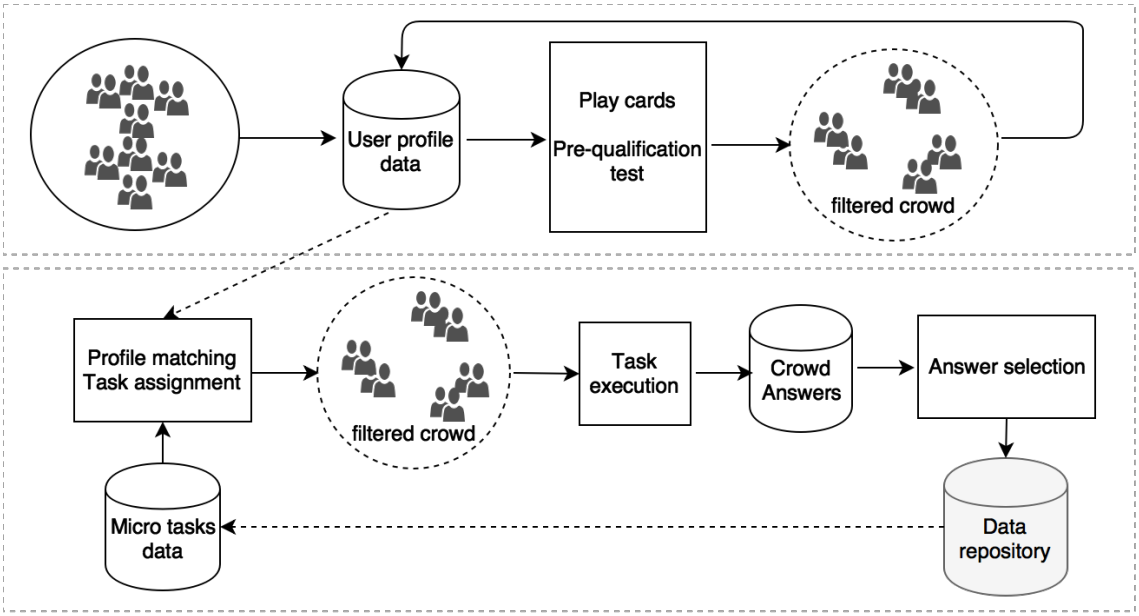


Figure 6.14 Quality control process - Qualification test for getting qualified

– *Year-Finder* - similar to Location-Finder, users are asked to provide the year when the image was taken.

– *Annotation-Competition* - users in pairs can compete in tagging images.

– *Validator* - users are asked to validate automatically generated tags and image categories [SSS20].

Gamification approaches are considered to incentivize the users participating in the tasks, as well as for data quality control.

6.3.3.7 Ensuring quality control for crowdsourced content

Crowdsourcing as a growing topic over the last decade has been applied to numerous domains, both in research and enterprise contexts. At the top of the list of issues that remain open and challenging in crowdsourcing are quality control and motivation. Here we present the implemented method which explores the use of gamification in crowdsourcing settings, as a means to: improve task assignment and performance, incentivize people to participate, and control the quality of their work.

To assess the quality of posted contributions by *City-Stories* platform participants, and to make the crowdsourcing tasks more attractive and engaging, we apply a game-based quality control mechanism that considers users' profiles and their interests. This process is illustrated in Figure 6.14. The *play-cards*

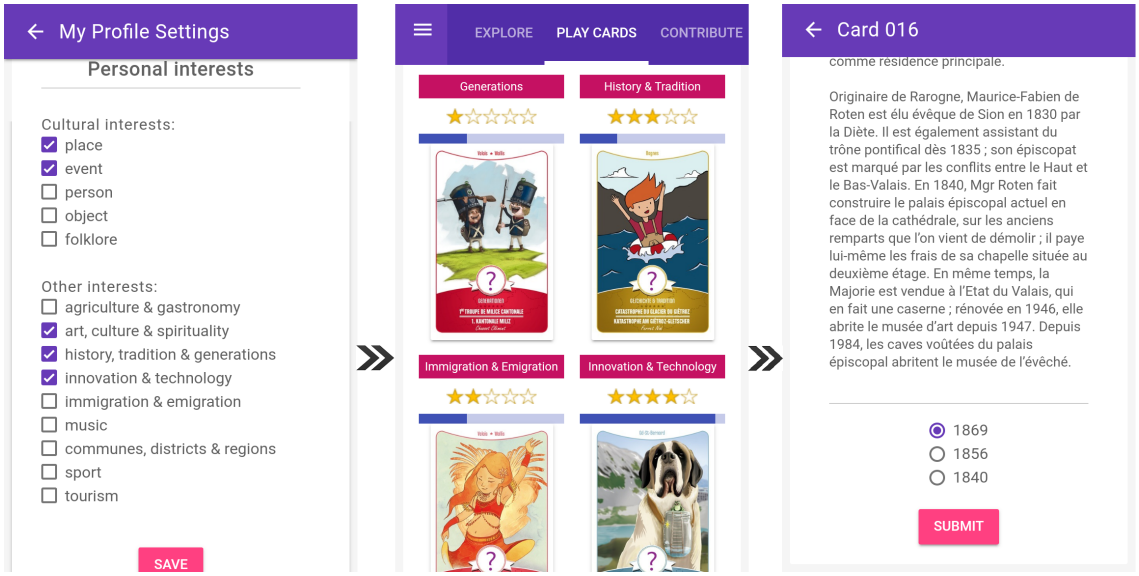


Figure 6.15 Quality control through gamification

game adapts game-design elements in a non-gaming context. It acts as a pre-qualification test for users who are motivated to annotate data, similar to *gold questions* [OSL+11]. This test consists of 195 cards grouped into 13 categories, and each one of these cards has a story behind it.

Initially, users provide information about their profile, especially information about their interests which fall within one or more of these 13 categories. Depending on this information, they are forwarded to read and answer questions/cards related to their predefined interests, hence avoiding unfair exclusion of workers due to non-relevant questions. For instance, a user that has chosen *sport* as his area of interest, he is asked to answer questions within that topic. He chooses to read the story of the cards and tries to guess the year which is related to the topic. To boost the motivation of users, two joker cards can be used. If the user successfully answers 70% of the cards, he is considered later as a potential worker for solving micro tasks related to that category. User's performance is visually displayed: in each category, the accuracy is shown in the form of stars and a progress bar shows the completeness of the questions of that category (depicted in Figure 6.15).

To further motivate qualified users to participate in data annotation, we apply a reputation mechanism. Depending on their level of contribution, users gain reputation points and titles. Top contributors will have the chance to receive public recognition from the archival institutions.

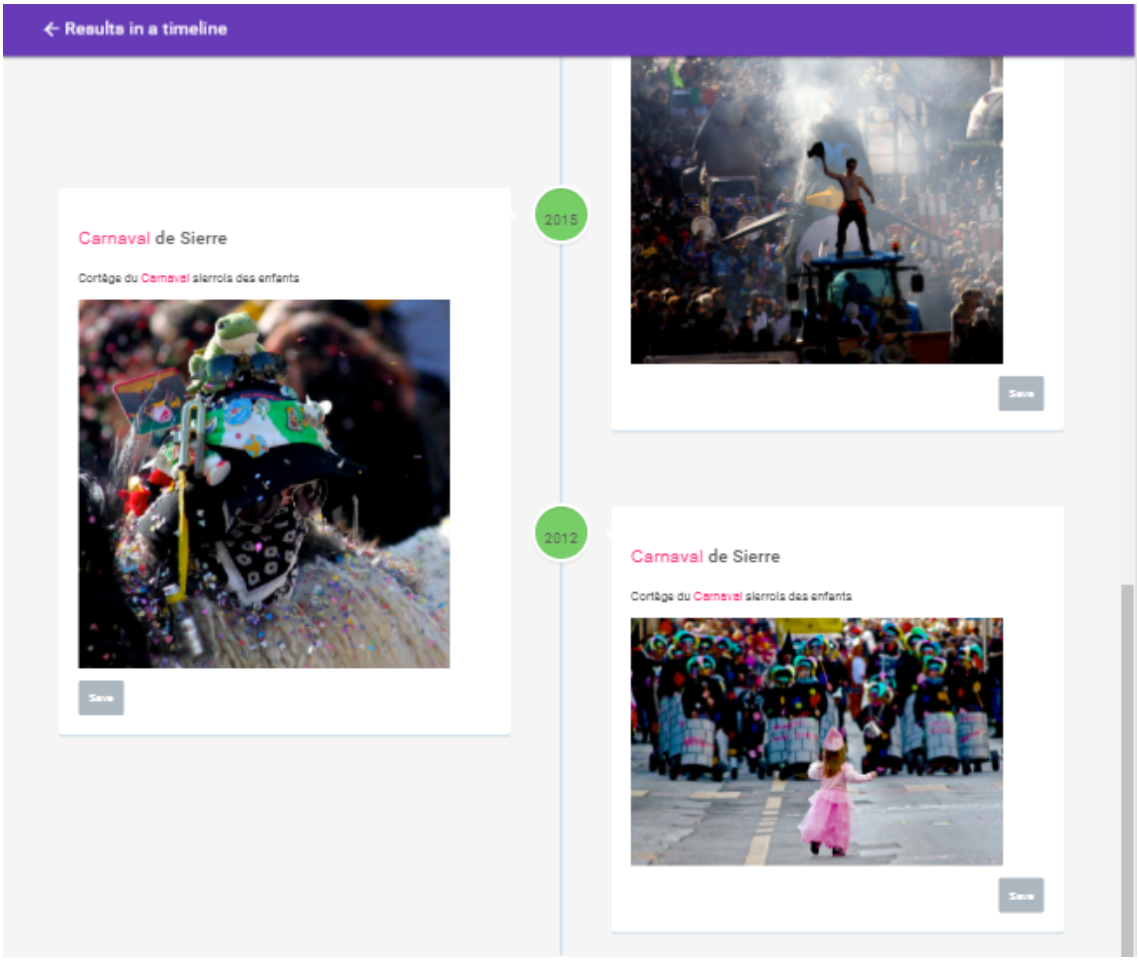


Figure 6.16 Timeline visualization of retrieved historical data [SLS18]

6.3.3.8 Semantic Network Visualization

In cultural heritage, quick access to the right information has an important role. Visualizing the data in a meaningful representation is a more recent concept used for accessing data, deriving meaning, and acquiring knowledge from the data [NBG+15]. Data visualization has an incredible power to attract people's attention, consequently, it enables users to derive concrete conclusions.

Here, we describe a method of visualizing the historical data to enhance the user's search experience through semantic search [SLS18]. Semantic technologies aim to provide more visually appealing data, by enabling graphical representations of the semantically structured data. Furthermore, it enables meaningful relations of data entities. The meaningful and labelled clustering of data in the form of semantic concepts enables new ways to visualize data. As a motivating scenario, we consider the following use case: a user, visiting a historical and tourism attraction site in Valais, uses the City-Stories application to find meaningful relations of data entities about Zermatt. Based on his search,

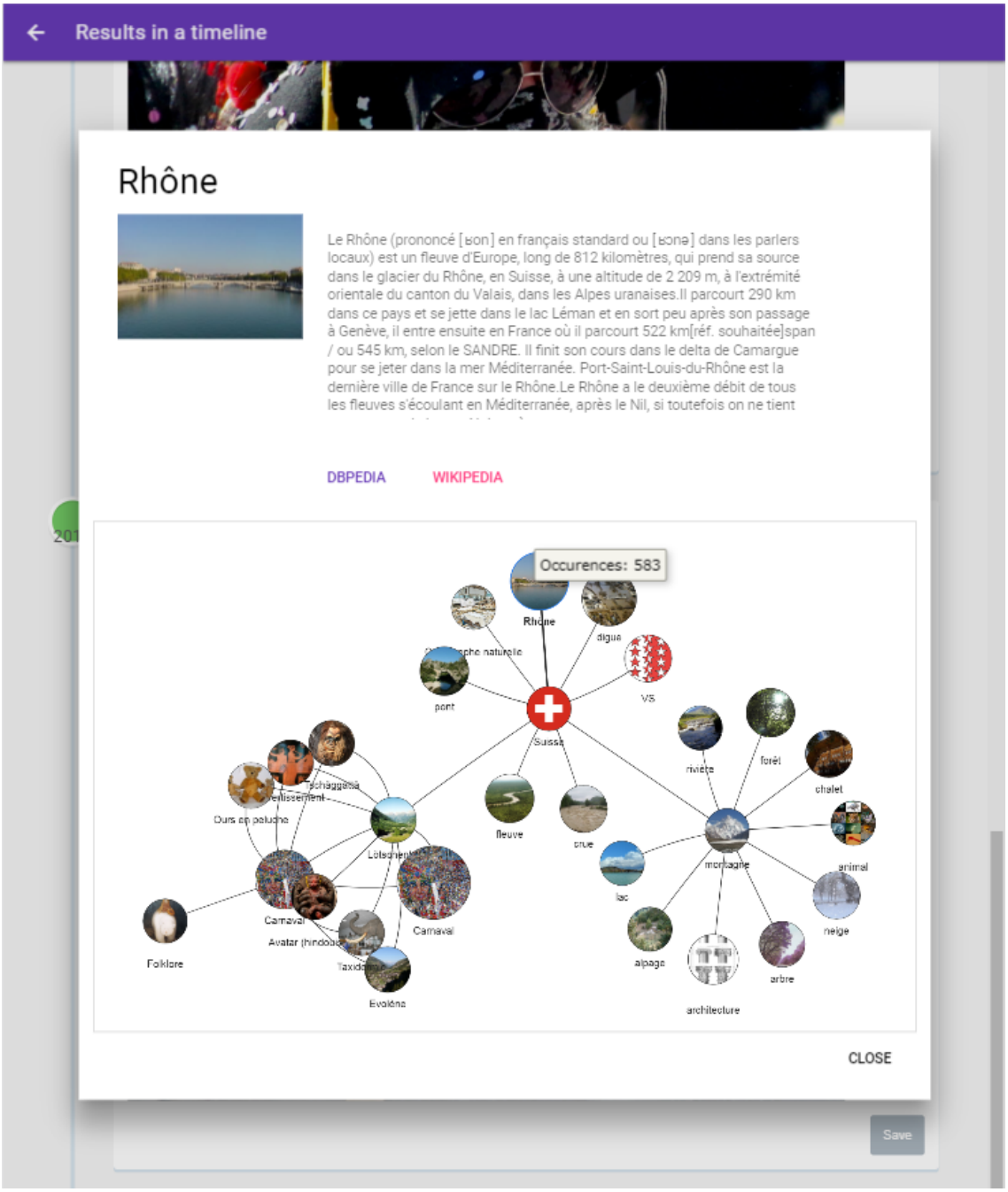


Figure 6.17 Network visualization of related entities around the browsed concept [SLS18]

the application recommends to the user the relevant information over Zermatt, and it visually displays the most important concepts such as related locations, people, events, and historical sites.

Because the dataset is specific and contains data about a region in Switzerland in local languages, and due to the short available text description in the media metadata, searching and finding relevant information is limited to key-

words strict matching. Hence, we add external knowledge base information (if available) in three languages. In this way, we enable the *semantic text search*, i.e. the text search keywords are mapped to the multilanguage linked data available in the dataset. Since the *City-Stories* datasets contain multimedia items mainly about a location, person, or event, the data is aggregated based on the title of the items, which is a more general representation and does not include specific information. The search results are *timeline visualized*, giving users a better experience in exploring how these places, things, and events evolved chronologically. An example of the timeline visualization is depicted in Figure 6.16, in which the user is searching for *"Carnival of Sierre"*, and images and videos are visually represented and ordered by time.

Extracted concepts within the text description of the multimedia items are highlighted in distinguishable colours. On the same page, users can read more about these concepts: images, descriptions, and links to the corresponding Wikipedia and DBpedia webpages are provided. The main part of the visualization is the *network of connected concepts*. The network consists of top k nodes and edges, where the nodes represent the strong concepts related to the chosen concept.

An example is illustrated in Figure 6.17, where the user is interested to learn more about "Sierre". On the user interface (UI) the network of 10 connected concepts is generated around the chosen concept "Sierre". The size of the circular nodes depends on the importance, i.e. number of occurrences of the concepts with respect to the chosen concept. By clicking on each node, users can read more about other concepts in the network. Additionally, exploring more information about the displayed multimedia is available, allowing users to expand and collapse the network.

Furthermore, users have the capability to visualize the connected concepts' trends over time in a network. This feature allows the exploration of the changes in important and relevant elements related to the chosen place, event, or person of interest. This is shown in Figure 6.18.

6.3.4 Overview of Experiment Dataset

A subset from the *NotreNistoire* dataset described in Section 6.3.4 was used for evaluating the hybrid join model. In this scenario, we only used images that had a "by-nc-nd" Creative Commons licence[17]. This is important in order to reproduce the experiments and validate the results we obtained. Finally, the dataset contains 5,015 images and metadata information about the images such

[17] https://creativecommons.org/licenses/by-nc-nd/2.0/

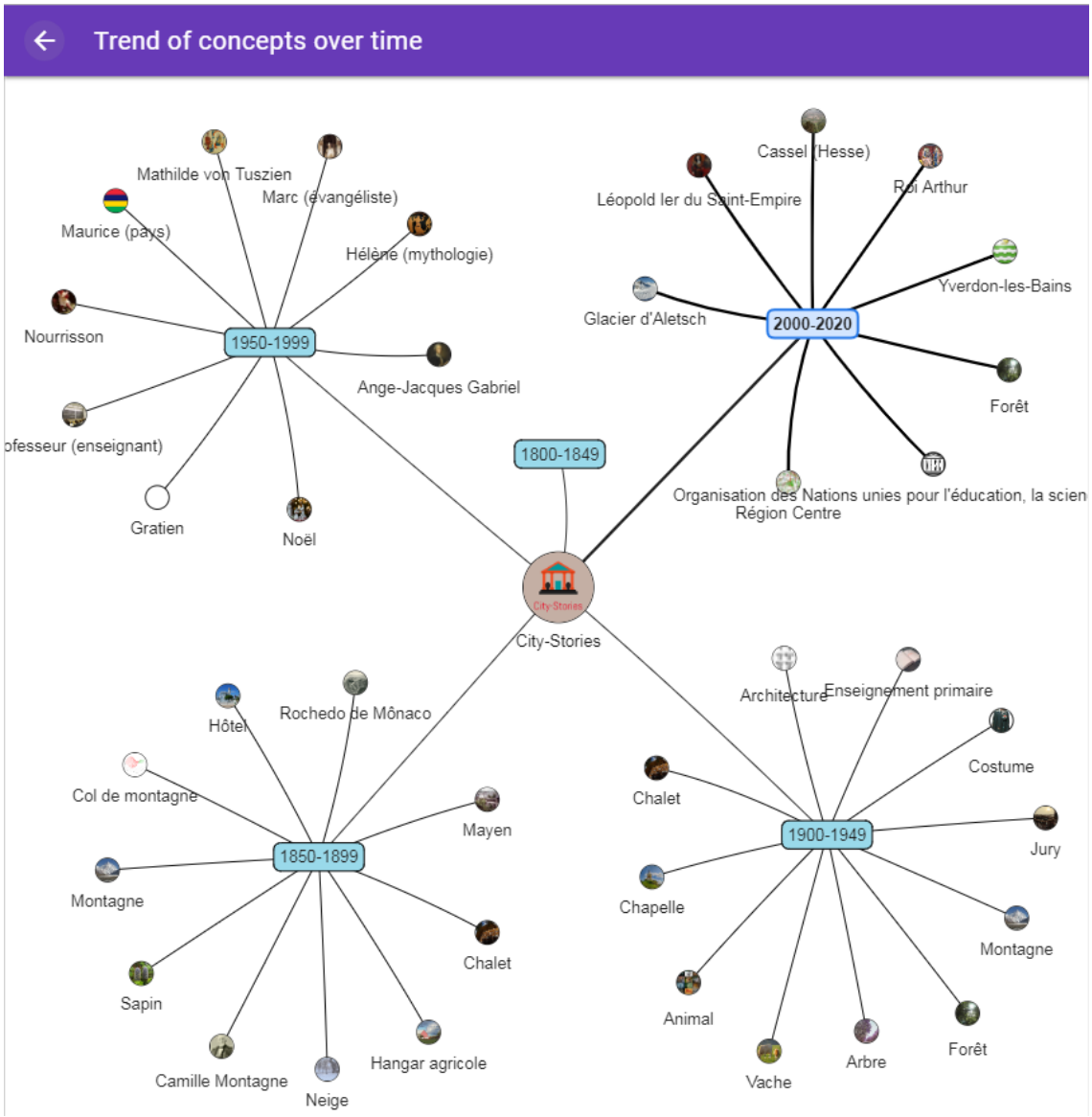


Figure 6.18 The overall network visualization of extracted concepts from the text in metadata [SLS18]

as title, description, location, year, tags, and author information. Figure 6.11 illustrates some example images from the dataset.

6.3.5 Pipeline

Below, we describe the end-to-end pipeline applied for the categorization of historical data. At the beginning, we describe our method for annotating the images to obtain ground truth labels. Next, we detail the implementation of the deep learning based approach for image categorization, followed by the crowdsourcing approach.

Figure 6.19 Web annotation tool

6.3.5.1 Image Annotation

The aim of this work is to enrich the metadata of the dataset with a new attribute *cultural_interest* by assigning each image one or more of the following categories:

1. *place* – if the image is showing a place/location (e.g., landscape, mountain, city view)

2. *person* – if the main theme of the image is a person or group of people and people are clearly identified in the image (e.g., portrait of people)

3. *event* – if the image depicts an organized event (e.g., carnival, festival)

4. *object* – if the image shows an object (e.g., sculpture, painting, specific vehicle, or building)

5. *tradition* – if the image shows people with specific clothes in events or performing particular activities

Table 6.9 Number of images that contain each category

Category	Number of images	Percentage
place	3,197	63%
object	1,742	35%
person	1,098	22%
event	515	10%
tradition	229	5%

Since the dataset was not labelled, ground-truth data was needed in order to evaluate the performance of automatic classification, crowdsourcing, and hybrid human-machine approaches. For this reason, we organized an annotation task by inviting 10 participants from the region. Participants were presented with the idea behind the project and the objectives. They were shown detailed instructions with comprehensive examples of how to annotate the image. In two rounds, the 10 trained annotators used a developed web annotation interface tool to annotate the images (Figure 6.19). In the first round, they used a pooling mechanism where each annotator was assigned an image that had not been annotated. In the second round, annotators were assigned images that had already been annotated by other annotators from the first round. Inter-rater agreement was $\kappa = 0.55$, and considering that it is a multi-label task, this number can be considered substantial [McH12]. An additional, third round was required to resolve annotators' disagreements. Table 6.9 represents the number of images that contain each of the five categories. Clearly, the most frequent category in the images is *place* which appears in 63% of the images, followed by the categories *object* and *person* with 35% and 22%. The categories *event* and *tradition* are the least represented categories in the dataset with 10% and 5%, respectively. Table 6.10 shows the number of categories per image. A single category appears in 3,382 or 67% of images, 1,502 images or about 30% have two categories, 129 images or about 3% contain three categories, only 2 images result to have four categories, and there was no image that had all five categories.

Table 6.10 Distribution of categories over the image dataset

#categories	one	two	three	four	five
number of images	3,382	1,502	129	2	0
percentage	67.44%	29.95%	2.57%	0.04%	0%

6.3.5.2 Automatic Classification via Transfer Learning

Thanks to the high number of images available online, image classification and object detection are some of the areas where deep learning has shown promising results. A deep neural network trained on enough large datasets can classify images with high accuracy. For instance, the ImageNet project [DDS+09] has a very large database of 14 million hand-annotated images that contain more than 20,000 classes. However, the process of collecting and annotating a dataset is expensive and time-consuming. Moreover, developing a new model from the ground up every time on small training data does not provide high accuracy. Therefore, pre-trained models and transfer learning [PY09] techniques reduce the effort needed to collect massive amounts of training data.

Pre-trained models are trained in the context of large and general classification tasks. Therefore, they can be used to address a more specific task by extracting and transferring meaningful features that were previously learned. Some of the popular image pre-trained models are VGG19[SZ14], MobileNet[HZC+17], and ResNet[SIV+17]. We use the MobileNet deep neural network and implement transfer learning with a fine-tuning method. The customized implementation has a logistic regression final layer with a sigmoid activation function and uses the binary cross entropy loss [ONB+17]. The last predicting layer of the pre-trained model is removed and replaced with the custom predicting layer that contains the five categories: person, object, place, event, and tradition. The multi-label classification model for each of the images assigns an image to one or more classes.

In parallel to building a model that relies on visual features, we investigate the importance of text information available in the existing metadata attributes. Our method uses an image labelling model to get image tags, entity extraction to find concepts, and DBpedia knowledge base to query concept categories. Our assumption is that the DBpedia categories can provide more specific information that connects directly with the image category we aim to predict. This is better illustrated in Figure 6.20. The example image is taken during the *Geneva Festival* and the title of the image is *Fetes de Geneve defile fanfare 1968*. The pre-trained vgg model additionally provides the tags “marching band”, “parade uniform” in English. The entity extraction tool recognizes the concepts “Fetes de Geneve”, “fanfare”, “marching band” and “parade”. Exploring the DBpedia category for the “Fetes de Geneve” concept gives us the “Fete Suisse” which is another event. On the other hand, the categories of “marching band” and “parade” give more context that this event is a tradition as well.

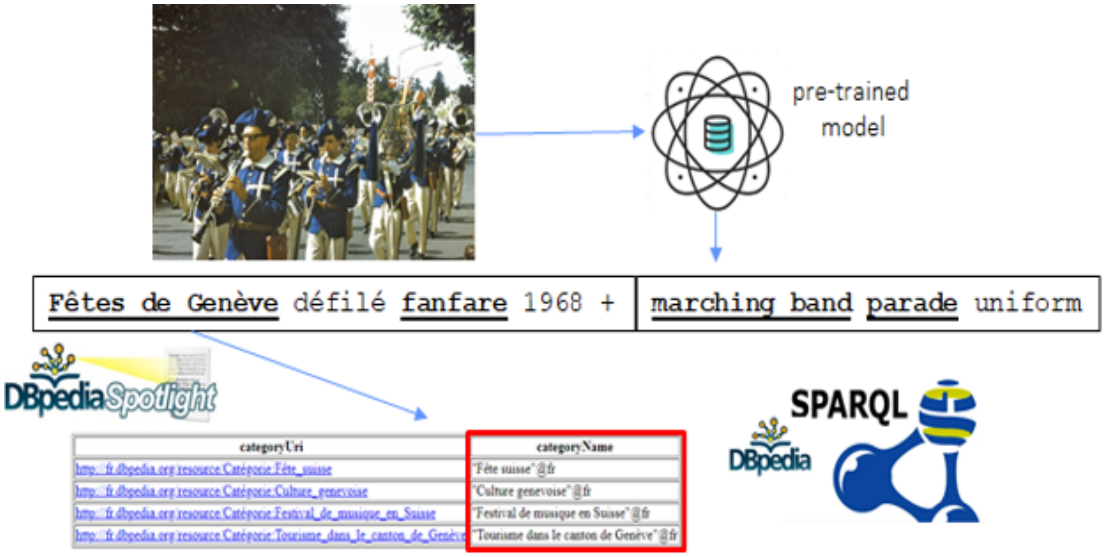


Figure 6.20 Extracting concepts and DBpedia categories

6.3.5.3 Crowdsourcing Approach

Collecting annotated data is an expensive and time-consuming process. Crowdsourcing has been widely used as an alternative service to replace experts with specific domain knowledge for labelling. It efficiently reduces the costs and latency by making use of the collective intelligence of thousands of available crowd users on the Internet. There are many popular non-paid crowdsourcing projects in citizen science [Han10] such as Wikipedia, GalaxyZoo [COS+15], and Recaptcha [AD08]. In parallel, there are several popular commercial crowdsourcing online platforms such as Amazon Mechanical Turk (MTurk)[18], FigureEight[19], MicroWorkers[20], etc. These platforms enable the exchange of HITs (Human Intelligence Tasks) between requesters who need tasks to be completed and workers who are available and willing to complete a task, and who get a financial reward for that work.

For categorizing the images of the dataset, we used the MicroWorkers crowdsourcing platform. Each image was used to generate a HIT, asking online crowd participants to categorize the image into one or more of the 5 classes. A HIT contained the URL of the image. Additionally, we added the title, location, and description of the image that could provide some helpful context. Crowd workers were instructed to analyse the image and provide the most suitable classes for that image. The five classification categories were place, object, per-

[18] https://www.mturk.com/
[19] https://figure-eight.com/
[20] https://microworkers.com/

son, event, or tradition. As a quality control mechanism, the [OSL+11] technique was applied. We used a set of qualification tasks to allow only qualified users who pass the test with an accuracy of at least 60% to keep on labelling the images. Additionally, reputation mechanisms [DAP15] were enabled, opening the annotation job only to the *best microworker group* that has the workers with the highest reputation on the platform. Asking multiple crowd workers to perform the same task is used, usually to increase the quality of the data by aggregating the answers. Therefore, for each image, we asked three crowd users to provide the answer. Depending on the task, sometimes even a simple Majority Voting (MV) aggregation algorithm increases the data quality.

Considering that in our case we have a multi-label task, having multiple judgments can lead to higher disagreement between annotators, yielding low-quality answers. Therefore, we applied three truth inference algorithms to infer the correct answer from the workers' answers. We decompose the worker answers into binary form. For instance, if a worker answer is *place* and *object*, from the set of classes ["place"', "object", "person", "event", "tradition"], the answer is encoded into a binary vector $[1, 1, 0, 0, 0]$. First, we consider the MV algorithm which simply selects as final the answer given by the majority of workers. Next, we evaluate the Dawid and Skene model [DS79] which is based on the Expectation-Maximization (EM) principle to model the worker's reliability with a confusion matrix for the answer aggregation. Last, we apply a truth inference algorithm that considers prior information provided by the crowdsourcing platform about a worker's profile. We derive the reputation of a worker based on his previous finished tasks (number of accepted and rejected tasks, money, and badges earned) and integrate that score in a weighted aggregation method to infer the true answer.

6.3.6 Hybrid Human-Machine Classification Architecture

While crowdsourcing reduces the cost and latency per annotation compared to domain experts, hybrid human-machine information systems aim to reduce the overall annotation costs by selecting only the most important instances for being annotated by humans.

The *high confidence switching* as a hybrid method only selects instances for which the machine learning model is uncertain. Current machine learning models also provide a confidence estimate [PNV+02] on how accurate their answer is. Therefore, predictions with low confidence values are considered further to be solved by crowd workers. This method is helpful in scenarios when the avail-

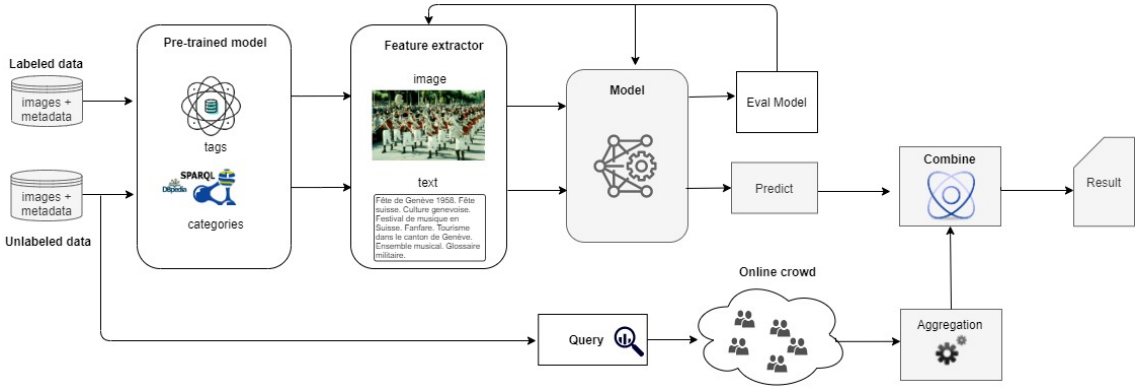


Figure 6.21 Overall architecture of the proposed hybrid human-machine classification model

ability of human annotators is limited and latency is critical, as it tries to boost the accuracy of automated algorithms by minimizing human input.

The *human-machine aggregation* is another hybrid method where the predictions of the machine learning model and the inferred crowd answers are jointly combined to resolve the final output. A multi-label classification task with multiple human annotators in some settings is prone to higher disagreement and a consensus on the output is not reached. On the other hand, for specific cases, machine learning models are not able to depict the context as good as humans. This method aims at combining weak responses to eventually increase the quality of results. The assumption is that the fusion in the aggregation will cancel eventual individual weaknesses. The *human-machine aggregation* method is suitable, especially for scenarios when latency and cost of classification are not an issue, but accuracy is essential. We experience this situation with the Notre-Histoire platform. This project has been running for several years, and it has thousands of registered users and hundreds of active members. The members volunteer to share new historical content, and they actively participate in the data curation process. We consider a weighted aggregation, where the estimates of the deep learning and the crowd aggregation models are multiplied with different scores. Their weights are derived based on their individual classification accuracies on the validation set. The sum of the weighted estimates results in the joint predicted output. The individual class estimates higher than a threshold are taken as predictions. Figure 6.21 illustrates the full pipeline of the human-machine approach for image categorization.

6.3.7 Experiment Results

In this section, we outline the evaluation of the proposed methods for our multi-label image categorization problem on the NotreHistoire dataset. In traditional binary and multi-class classification problems, commonly used evaluation metrics are precision, recall, and the F1 score. However in multi-label classification tasks, there are additional evaluation metrics such as exact match accuracy (subset accuracy) and Hamming-loss. Exact match is a strict metric measuring the percentage of the samples that have all their labels classified correctly, whereas the Hamming-loss measures only the fraction of wrong labels to the total number of labels, thus penalizing the individual labels. We use these two metrics to evaluate the performance of the automatic classification based on visual and textual features, the crowdsourcing approach, and the hybrid human-machine method.

6.3.7.1 Automatic Classification

The starting point for the evaluation is to split the modelling data into training, validation, and testing sets. We decided to allocate 60% of the data for training, 20% for validating the models, and 20% for the test set. The original dataset classes are strings that are easy to understand by humans. However, to build and train a neural network model on a multi-label scenario, binary labels are generated from multi-hot encoding. Since the image dataset has images with metadata attached to them, we considered the available text from the metadata attributes: "title", "description", and "tags" as input text to the model. Additionally, from the concatenated text, we extracted labels with a pre-trained VGG16 model [SZ14]; extracted the DBpedia concepts with dbpedia-spotlight [MJGS+11]; and the DBpedia categories of each extracted concept were retrieved with the DBpedia SPARQL endpoint. Finally, each image metadata has the title, description, user-provided tags (if available), automatically extracted labels, DBpedia concepts, and the categories of these concepts.

Initially, we evaluate the accuracy of the machine learning models by considering only textual features. Feature extraction is run on the final combined text by using the term frequency, inverse document frequency (tf-idf) with the following parameters:

- $min\_df = 3$
- $max\_features = 3000$

- $stop\_words$ = English + French
- use_idf = 1, and
- $analyzer$ = word

. After that, we selected three different machine learning classification models: i.) Logistic Regression (Log), ii.) Random Forest (RF), and iii.) Support Vector Machines (SVM). Table 6.11 presents the classification results based only on textual features. While the accuracies of the three models are similar, the SVM model achieved the highest accuracy of 58% on the testing set.

To analyse the importance of the generated features, we run the evaluation of the models separately based on different settings. We consider the following combinations of text available in the metadata and the additional generated data:

(i) title + description *(TD)*

(ii) title + description + vgg labels *(TDV)*

(iii) title + description + vgg labels + dbpedia-spotlight entities *(TDVE)*

(iv) title + description + vgg labels + dbpedia-spotlight entities + dbpedia categories *(TDVEC)*

Table 6.12 shows the accuracy of each model when using features extracted on the combinations mentioned above. Considering only the originally provided text on the metadata of the images (TD), SVM achieves an accuracy of 51% and a Hamming-loss of 12%, whereas the RF and Log models achieve an accuracy of 49% and Hamming-loss of 12%. Our initial assumptions that adding more information will boost the accuracy were valid, and this can be observed from the results shown in Table 6.12. All three models perform better when more text information is added to the input. The text provided by the TDVEC combination reached the highest accuracy of 58%.

Our next step was to evaluate the performance of a deep learning multi-input model that combines text and image features. Considering that our dataset

Table 6.11 Accuracy of machine learning models according to textual features

Classifier	Accuracy	Hamming-loss
Logistic Regression	56%	13%
Random Forest	57%	12%
Support Vector Machines	58%	12%

Table 6.12 Accuracy of models with different combination of text data

Combination	Logistic	Random Forest	SVM
TD	49%	49%	51%
TDV	51%	52%	54%
TDVE	55%	56%	57%
TDVEC	**56%**	**57%**	**58%**

consists of 5,015 samples, the size of this collection would not be large enough to build and train a deep learning model from scratch. Therefore, we apply transfer learning techniques by using the MobileNet [HZC+17] and GloVe [PSM14] pre-trained models. MobileNet model was set to use weights from ImageNet which is trained on a large image collection and with a more general classification task. We configured it with a depth multiplier of 1.0 and an input size of 224×224. A new classifier with our custom dataset labels on top of it was added. Accordingly, the dataset images were resized to adapt to the input expected by the MobileNet model. In the text input, we added an embedding layer with loaded weights from the GloVe pre-trained word embeddings.

The new custom classification head was trained with images of our dataset (training set) so that the model addresses the multi-label classification task. A learning rate of 1e-5 on the training process was used, and the performance on the validation set was measured on 30 epochs. Figure 6.22 outlines the training and validation accuracy score of the multi-input deep learning approach. After 30 epochs, our model achieved an accuracy of 63% and Hamming-loss of 10% on the validation set. It is important to emphasize the effect that the transfer learning method has on the model's accuracy. During the first 10 epochs of training and the validation process, we set the layers of the MobileNet pre-trained model as non-trainable (frozen). After that, the last 100 layers (of the total 155 layers) were "unfrozen" and we retrained the model for another 20 epochs. We can observe that the accuracy increases (after the yellow vertical line) as an effect of transfer learning. Finally, we evaluated the model with data from the testing set, and the accuracy achieved was 62% with a 10% Hamming-loss.

6.3.7.2 Crowdsourcing Results

Considering that this is a multi-label task, we assume that lower-represented classes such as "event" and "tradition"" are more challenging for a model to predict. In contrast to this, humans have the potential to perform better, especially

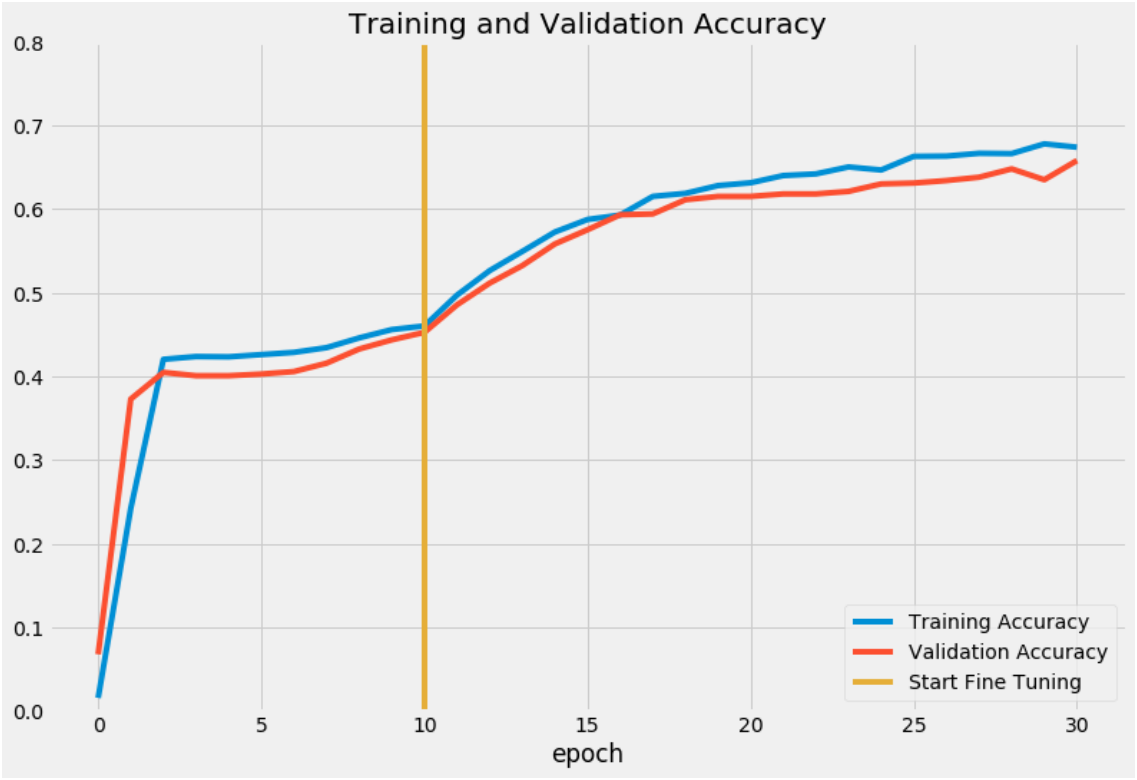


Figure 6.22 Training and validation accuracy on multi-input text and image model

when identifying if an image is from the context of "tradition" or "event". As a result, crowdsourcing has been considered an alternative solution to this image categorization problem. For the same categorization task, we thus make use of the MicroWorkers crowdsourcing platform to generate HITs, asking online crowd workers to categorize each image from the dataset. The entire set of images was used to generate 5,015 HITs, one HIT per image. The original metadata of the image collections are in French, and automatic classification models use this format. However, for the online crowdsourcing task, we translated the text automatically to English to expose the task to the largest crowdworker groups on the platform who speak English. Task design techniques [FKT+13] are essential factors that increase the quality of crowdsourced data. Therefore, elements such as instructions, rules, tips, and examples have been thoroughly considered in our project to guide and help the online workers solve the tasks. Furthermore, a reputation-based method was applied, by opening the job only to the "best annotators" group of the platform. For each image, we asked three workers to provide the categories. The annotation by online crowd users took 8 hours effective time to complete, with 542 participants from 78 countries on average completing 31 tasks. Since our goal is to compare the accuracy of crowdsourc-

Table 6.13 Results from crowdsourced data aggregation methods, deep learning, and hybrid human-machine method

Method	Accuracy	Hamming-Loss
#1 worker	49%	14%
#2 workers	47%	14%
#3 workers	41%	16%
Majority Voting	56%	12%
Dawid-Skene	58%	12%
Worker-Profile	**58%**	**11%**
Deep Learning	**62%**	**11%**
Hybrid human-machine	**65%**	**10%**

ing with the automated approach, we use the annotations from the validation set and testing set to compare the three different aggregation methods described in Section 6.3.5.3. Our assumption that in a multi-label task, redundancy can lead to higher disagreement was confirmed. If we simply aggregate the answers without any quality control mechanism, adding the responses from the second and the third annotator reduces the accuracy from 49% to 41%. Data quality is a major issue in crowdsourcing, therefore we evaluate the MV algorithm, the Dawid-Skene model, and our worker profile model. Table 6.13 details the results obtained from the crowdsourcing experiment. The MV algorithm decomposes the task in binary output where for each of the five classes (place, object, person, event, tradition) the majority vote is taken as a final answer. The majority voting achieves an accuracy of 56% and hamming-loss of 12%, the Dawid-Skene and worker-profile models achieve slightly higher accuracy of 58% whereas the latter one has lower hamming-loos of 11%.

6.3.7.3 Hybrid Human-Machine Image Categorization

So far, we have observed that text information from the image metadata, together with additional semantic information extracted from Wikipedia, can improve the classification accuracy of machine learning models. On the other hand, due to the multi-label task, disagreements between annotators resulted in lower accuracy of crowdsourced data. However, we expect that joining the outputs of the two approaches will improve the overall accuracy, complementing each other's strengths.

To evaluate the hybrid human-machine aggregation method described in Section 6.2.4.3, we use a weighted sum of the class estimates of the deep learning

model and the inferred classes from the answer aggregation of crowd workers. Since the automatic approach provided better results in general, we expect that weighting higher its output compared to the crowd answer will perform better. Therefore, the validation set was used to test different weight scores for the deep learning and crowd outputs. We found that the weights of 0.7 for deep learning and 0.3 for crowd outputs achieved the highest strict accuracy of 65% and hamming-loss of 10%. Incorporating the human judgments in the final output was shown to improve the accuracy by 3%. Since the image categorization is a multi-label task with five classes, this can be considered as an improvement.

6.3.8 Discussion

Automatically classifying the images by methods that used visual features and textual features achieved an accuracy of 58% and 62%, respectively. The reason why the multi-input deep learning model performs better is that the information about the extracted concepts and their categories gathered from DBpedia extends the context of the image item. This extension is especially helpful for the classes "event" and "tradition" since it is difficult to extract features that can distinguish between these two classes for an image. Although the overall improvement of 4% in accuracy and 1% for the hamming-loss is not high, it is nevertheless higher for the two less represented classes of the dataset ("event" and "tradition"). Such an example is illustrated in Figure 6.20.

On the other hand, we experienced that the highest accuracy achieved by aggregating crowd answers was 58%, which is lower than the accuracy of the deep learning model – however with the hamming-loss being the same. One reason for the lower accuracy is that for specific images there is a disagreement between annotators, especially when judging whether the image class is "place" and/or "object". In the answer aggregation, we note that majority voting does not require prior data, whereas the Dawid-Skene model used the training set to estimate the workers' reliability. In the worker-profile method, we did not use the training set, however, we relied on profile information provided by the MicroWorkers platform.

6.3.9 Experiment Summary and Limitations

In this experiment, we have addressed the task of categorizing images from cultural heritage collections. Enriching the images with additional information and maintaining high-quality metadata are key factors for finding them later on.

We presented our hybrid human-machine framework for image categorization. This method aggregates the predictions of a multi-input deep learning model and the inferred true annotations from multiple crowd users. The deep learning model uses visual features extracted from the images and features extracted from text in the image metadata. We found that our method of adding Wikipedia classes of the concepts extracted in the text improves the classification accuracy. Moreover, incorporating annotations from crowd users and applying a weighted aggregation additionally improved the results.

In summary, our results have confirmed the assumption that the hybrid aggregation method is an effective approach to combining machine learning with crowd annotation skills. This method is helpful for organizations like GLAMs that maintain data repositories in the cultural heritage domain, and which have many active participants. Currently, the NotreHistoire platform has thousands of registered users and several hundred active participants. Hence, the proposed strategy is well applicable to categorizing the images in such a context.

The size of the dataset is a possible limitation that especially affects the deep learning approach. In principle, deep neural network models require more data [SSS+17], therefore, our future work will focus on increasing the size of the dataset to improve the accuracy. We identified that the collection metadata has additional issues such as missing high-quality *image tags* and the *period* which defines the temporal decade of the images. We plan to deploy our method to address these additional missing data.

Part IV

Conclusion

7

Conclusion and Future Perspectives

This final chapter summarizes the objectives, contributions, and results behind this thesis. The underlying motivation and findings of this thesis shed light upon the emerging futuristic research area of *hybrid intelligence*, which combines *machine intelligence* (e.g., supervised machine learning) and *human intelligence* (e.g., crowdsourcing) in a new framework to overcome their limitations as standalone applications.

7.1 Summary

In this thesis, we have motivated and provided a vision on the use of hybrid intelligence methods as advantageous solutions for efficient and scalable data processing applications.

Starting with Chapter 1, we introduced the general problem (Section 1.2) upon which this thesis is built, the challenges of solving data-related problems with machine learning based and human based approaches individually. We have seen that the challenges fall in three dimensions: i) *accuracy* of the automated data processing approaches based on machine learning, ii) the *latency* caused when involving human intelligence which can provide better accuracy, and iii) the *cost* of crowd wisdom. The general problem was supported with exemplary scenarios illustrated in Section 1.3.

As this thesis proposes a hybrid intelligence methodology that combines machine and human intelligence components, the fundamentals of these two individual approaches were presented. In Chapter 2, the fundamentals of machine

learning with a focus on supervised learning methods were introduced. Considering that the more frequent applications of machine learning are largely related to text and image analysis, we described the concepts behind text classification methods, as well as image classification based on deep learning approaches. In the following Chapter 3, the fundamentals of the crowdsourcing research field were introduced, motivating the importance of human intelligence in generating high-quality data, complementing machine learning based approaches that fall short in providing highly accurate results. Here, the challenges associated with the sole application of crowdsourcing techniques were emphasized. Driven by the challenges that the aforementioned methods have individually, in Chapter 4, the motivation behind bridging these two techniques as the potential solution that overcomes these challenges was introduced. In a nutshell, Chapters 2 and Chapter 3 serve as the foundations upon which our proposed methods are built on, whereas Chapter 4 motivates the idea behind the hybrid human and machine information systems.

Subsequently, Chapter 5 introduces the benefits of the hybrid intelligence methodology and its modules: i) the data input, ii) the hybrid intelligence module, and iii) the data output. The hybrid intelligence component is the core part, as it consists of combining the machine intelligence and human intelligence subcomponents. The concepts behind the machine and human intelligence components rely on the foundation introduced in Part II, i.e., Chapter 2 machine learning and Chapter 3) crowdsourcing. After introducing the hybrid intelligence concept, three different hybrid human-machine models were presented: i) the human-in-the-loop model (HITL, Section 5.4), ii) high-confidence switching model (HCSM, Section 5.5), and iii) hybrid human-machine join prediction model (HMJP, Section 5.6). Each of the three proposed hybrid models was unique and applicable to solving various data processing problems. As stated above, the general problem lies between the three-dimensional issues: accuracy, latency, and cost. The three models (HITL, HCSM, and HMJP) are generic and address many data processing tasks that fall within this triangle. They address the fundamental trade-off challenges illustrated in scenarios in Section 1.3. In order to understand which model is suitable for application, a higher-level decision component was proposed. Depending on the nature of the given problem or task, one of the three models is chosen based on the evaluation of the three criteria: accuracy, latency, and cost, with respect to their importance.

Finally, in Part III (Chapter 6), the proposed hybrid intelligence methods were put into operation, conducting experiments for each method. These exper-

iments were motivated based on the scenarios described in Section 1.3, where we implement and evaluate the models with real-world datasets. Evaluations show that the proposed hybrid intelligence models outperform machine learning and crowdsourcing when applied individually. For instance, the experiment evaluating the HITL method in Section 6.1 showed that keeping the feedback of human intelligence in the loop of machine learning decision-making can boost the accuracy. This method is recommended for scenarios when accuracy is critical, while access to a large group of active crowd contributors is available. The second experiment described in Section 6.2 which implements and evaluates the HCSM model showed that introducing human feedback provides higher accuracy based on the trade-off threshold that decides when human input is needed. This method is useful for scenarios when all three criteria (accuracy, latency, and cost) have equal relevance. The third experiment presented in Section 6.3 showed that joining the decisions from both machine and human intelligence components results in higher accuracy, as the two parts complement each other, cancelling the individual weaknesses. Moreover, in this experiment, we showed that hybrid intelligence designs can be integrated into larger systems, as is the case with the *City-Stories* system. In a nutshell, the conducted experiments showed that hybrid intelligence models can address scalability by achieving higher accuracy while maintaining latency and cost.

7.2 Future Work

In the course of this research, new challenges and opportunities were identified. In the following, a few directions for future research are outlined.

7.2.1 Transparent Hybrid Joint Prediction Model

In Section 5.6, the Joint Prediction Model was presented. It combines the predictions from the machine intelligence (MI) and human intelligence (HI) components. The experiment conducted in Section 6.3 was about categorizing images in a multi-label task. Considering the triangle challenging criterion of accuracy, latency, and cost, this model treats the three criteria as equally important, therefore, every task is solved by both MI and HI components. However, the classifications from the MI component (joint deep learning model) are not known to the crowd contributors when providing their categories for each task. A future work would be to make the classification task *transparent* to the HI component. When

providing their feedback, human contributors would be shown the predictions of the MI component. An interesting perspective involves analysing the impact of transparency on the overall accuracy of the classification task. This analysis considers whether improvements in the predictions from the MI component enhance the results or introduce bias in the human feedback. Besides showing only the predictions, another scenario would be to show the confidence estimate, and how that impacts the decisions made by human contributions.

7.2.2 Explainable Hybrid Intelligence

While machine learning models have seen widespread adoption, explainability remains a challenge. Understanding fully the process from input to output, and what are the key reasons supporting the predictions made by the models, are essential for evaluating the trust in these models. In the last few years, extensive research around explainable machine learning [BP21] has been done, trying to fade the perception of "black boxes" for these models. For instance, LIME [RSG16] is a technique that explains the classifiers' predictions in an interpretable manner, providing representative features as an explanation of the decision.

An extension work on the transparent joint prediction model where the predictions from the machine intelligence component are presented to the human intelligence component, in addition to the decision, the machine learning algorithms will provide an explanation such as which features are and are not relevant for the decision. On the other hand, human feedback will review the explanation in terms of features (e.g., remove features that the model considers important, and vice versa, add features that are considered non-relevant). Furthermore, the human component can provide additional features that are not part of the decision analysis (e.g., in text classification, words that are part of the input text, or in image classification, segments of the image). In the same spirit as explainable ML, the decision on the human component part would be explainable, justifying its prediction. Hence, explainability is dual, leading to an explainable hybrid intelligence model.

7.2.3 Label Validator: A Hybrid Intelligence Model

The success of supervised machine learning in various domains depends on high-quality labelled datasets. These large datasets are critical to developing and training models deployed in fields such as image, text, and audio classification.

Machine learning models are as good as the data used to train them. Due to the large size of the datasets, crowdsourcing has been widely used for labelling. However, in Section 3.2, as one of the challenges in the crowdsourcing field, we listed the data quality and control mechanisms. Especially in large data labelling campaigns, errors occur. More recent work [NAM21] focused on finding errors in large labelled datasets and found that model accuracies on corrected test sets are different from the erroneous datasets.

In this spirit, a hybrid human-machine model could be deployed to continuously evaluate and validate generated dataset labels. Considering that re-evaluation or re-labelling via crowdsourcing is an expensive task, this model would iteratively evaluate chunks of the dataset, identify erroneous labels, re-train the models and thus increase the accuracy to evaluate and identify errors in other chunks of the dataset.

Bibliography

[AW10] Hervé Abdi and Lynne J. Williams. Principal component analysis. en. *WIREs Computational Statistics*, 2(4):433–459, 2010. ISSN: 1939-0068. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wics.101.

[AZ12] Charu C. Aggarwal and ChengXiang Zhai. A Survey of Text Classification Algorithms. en. In Charu C. Aggarwal and ChengXiang Zhai, editors, *Mining Text Data*, pages 163–222. Springer US, Boston, MA, 2012. ISBN: 978-1-4614-3223-4.

[AAM+17] Munir Ahmad, Shabib Aftab, Syed Shah Muhammad, and Sarfraz Ahmad. Machine learning techniques for sentiment analysis: A review. 8(3):27, 2017.

[AHC19] Sajjad Ahmed, Knut Hinkelmann, and Flavio Corradini. Combining machine learning with knowledge engineering to detect fake news in social networks-a survey. In *AAAI'19 spring symposium*, 2019.

[ALSG17] Tanja Aitamurto, Hélène Landemore, and Jorge Saldivar Galli. Unmasking the crowd: participants' motivation factors, expectations, and profile in a crowdsourced law reform. *Information, Communication & Society*, 20(8):1239–1260, August 2017. ISSN: 1369-118X. Publisher: Routledge _eprint: https://doi.org/10.1080/1369118X.2016.1228993.

[ABI+13] Mohammad Allahbakhsh, Boualem Benatallah, Aleksandar Ignjatovic, Hamid Reza Motahari-Nezhad, Elisa Bertino, and Schahram Dustdar. Quality control in crowdsourcing systems: Issues and directions. *IEEE Internet Computing*, 17(2):76–81, 2013. Publisher: IEEE.

[AG17] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. Working paper 23089, National Bureau of Economic Research, 2017. Series: Working paper series.

[AVGR+21] Miguel A. Alonso, David Vilares, Carlos Gómez-Rodríguez, and Jesús Vilares. Sentiment Analysis for Fake News Detection. en.

Electronics, 10(11):1348, January 2021. Number: 11 Publisher: Multidisciplinary Digital Publishing Institute.

[Alo13] Omar Alonso. Implementing crowdsourcing-based relevance experimentation: an industrial perspective. en. *Information Retrieval*, 16(2):101–120, April 2013. ISSN: 1573-7659.

[ABB+19] Saleema Amershi, Andrew Begel, Christian Bird, Robert DeLine, Harald Gall, Ece Kamar, Nachiappan Nagappan, Besmira Nushi, and Thomas Zimmermann. Software Engineering for Machine Learning: A Case Study. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, pages 291–300, May 2019.

[BMG+12] Yoram Bachrach, Tom Minka, John Guiver, and Thore Graepel. How to grade a test without knowing the answers: a Bayesian graphical model for adaptive crowdsourcing and aptitude testing. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, pages 819–826, Madison, WI, USA. Omnipress, June 2012. ISBN: 978-1-4503-1285-1.

[BYRN11] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval: The concepts and technology behind search*. Addison-Wesley Publishing Company, USA, 2nd edition, 2011. ISBN: 978-0-321-41691-9.

[BGL+16] Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. Research-paper recommender systems: a literature survey. en. *International Journal on Digital Libraries*, 17(4):305–338, November 2016. ISSN: 1432-1300.

[BBF18] Abdelhak Belhi, Abdelaziz Bouras, and Sebti Foufou. Leveraging known data for missing label prediction in cultural heritage context. *Applied Sciences*, 8(10):1768, 2018. Publisher: Multidisciplinary Digital Publishing Institute.

[BP21] Vaishak Belle and Ioannis Papantonis. Principles and Practice of Explainable Machine Learning. *Frontiers in Big Data*, 4, 2021. ISSN: 2624-909X.

[Bis13] Christopher M. Bishop. *Pattern Recognition and Machine Learning: All "just the Facts 101" Material*. en. Springer (India) Private Limited, 2013. ISBN: 978-81-322-0906-5. Google-Books-ID: HL4HrgEACAAJ.

[Bla04] Sarah Blakeslee. The CRAAP test. en. 31(3):4, 2004.

[BSW+19] David Bourgeois, James Smith, Shouhong Wang, and Joseph Mortati. Information Systems for Business and Beyond. *Open Textbooks*, January 2019.

[BF17] Petter Bae Brandtzaeg and Asbjørn Følstad. Trust and distrust in online fact-checking services. *Communications of the ACM*, 60(9):65–71, 2017. Type: Journal article.

[BFD18] Petter Bae Brandtzaeg, Asbjørn Følstad, and Maria Angeles Chaparro Dominguez. How journalists and social media users perceive online fact-checking and verification services. *Journalism Practice*, 12(9):1109–1129, 2018. Type: Journal article.

[BMS+18] Joel Breakstone, Sarah McGrew, Mark Smith, Teresa Ortega, and Sam Wineburg. Why we need a new approach to teaching digital literacy. *Phi Delta Kappan*, 99(6):27–32, March 2018. ISSN: 0031-7217. Publisher: SAGE Publications Inc.

[CGM+18] CallaghanWilliam, GohJoslin, MoharebMichael, LimAndrew, and LawEdith. MechanicalHeart. EN. *Proceedings of the ACM on Human-Computer Interaction*, November 2018. Publisher: ACM PUB27 New York, NY, USA.

[CLP+18] Francesco Cappa, Jeffrey Laut, Maurizio Porfiri, and Luca Giustiniano. Bring them aboard: Rewarding participation in technology-mediated citizen science projects. en. *Computers in Human Behavior*, 89:246–257, December 2018. ISSN: 0747-5632.

[CCR15] Yimin Chen, Niall J. Conroy, and Victoria L. Rubin. News in an online world: The need for an "automatic Crap Detector
. In *Proceedings of the 78th ASIS&T annual meeting: Information science with impact: Research in and for the community*, ASIST '15, 81:1–81:4, Silver Springs, MD, USA. American Society for Information Science, 2015. ISBN: 0-87715-547-X. Number of pages: 4 Place: St. Louis, Missouri tex.acmid: 2857151 tex.articleno: 81.

[CCH+18] Zhaoqiang Chen, Qun Chen, Boyi Hou, Murtadha H. M. Ahmed, and Zhanhuai Li. Improving the Results of Machine-based Entity Resolution with Limited Human Effort: A Risk Perspective. *CoRR*, abs/1805.12502, 2018. arXiv: 1805.12502.

[CMI+15] Xu Chu, John Morcos, Ihab F. Ilyas, Mourad Ouzzani, Paolo Papotti, Nan Tang, and Yin Ye. KATARA: A data cleaning system powered by knowledge bases and crowdsourcing. In *Proceedings of the 2015 ACM SIGMOD international conference on management of data*, SIGMOD '15. ACM, 2015.

[Cia18] Giovanni Luca Ciampaglia. Fighting fake news: a role for computational social science in the fight against digital misinformation. *Journal of Computational Social Science*, 1(1):147–153, 2018.

[CLF+20] Celia Cintas, Manuel Lucena, José Manuel Fuertes, Claudio Delrieux, Pablo Navarro, Rolando González-José, and Manuel Molinos. Automatic feature extraction and classification of Iberian ceramics based on deep convolutional networks. *Journal of Cultural Heritage*:106 –112, 2020.

[Co02] ICOMOS International Cultural Tourism Committee and others. International cultural tourism charter: Principles and guidelines for managing tourism at places of cultural and heritage significance. 13 june 2013, 2002.

[CRC15] Nadia K. Conroy, Victoria L. Rubin, and Yimin Chen. Automatic deception detection: Methods for finding fake news. en. *Proceedings of the Association for Information Science and Technology*, 52(1):1–4, 2015. ISSN: 2373-9231. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/pra2.2015.145052010082.

[CKT+10] Seth Cooper, Firas Khatib, Adrien Treuille, Janos Barbero, Jeehyung Lee, Michael Beenen, Andrew Leaver-Fay, David Baker, Zoran Popovic, and Foldit Players. Predicting protein structures with a multiplayer online game. *Nature*, 2010.

[COS+15] Joe Cox, Eun Young Oh, Brooke Simmons, Gary Graham, Anita Greenhill, Chris Lintott, Karen Masters, and James Woodcock. Doing good online: An investigation into the characteristics and motivations of digital volunteers. *Leeds University Business School Working Paper*, (16-08), 2015.

[Cro19] W. Bruce Croft. The Importance of Interaction for Information Retrieval. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, pages 1–2, New York, NY, USA. Association for Computing Machinery, July 2019. ISBN: 978-1-4503-6172-9.

[CL20] Limeng Cui and Dongwon Lee. CoAID: COVID-19 healthcare misinformation dataset, 2020.

[DZ14] Mita K Dalal and Mukesh A Zaveri. Opinion mining from online user reviews using fuzzy linguistic hedges. *Applied computational intelligence and soft computing*, 2014, 2014. Publisher: Hindawi.

[DAP15] Maria Daltayanni, Luca de Alfaro, and Panagiotis Papadimitriou. WorkerRank: Using employer implicit judgements to infer worker reputation. In *Proceedings of the eighth ACM international conference on web search and data mining*, 2015.

[DKC+18] Florian Daniel, Pavel Kucherbaev, Cinzia Cappiello, Boualem Benatallah, and Mohammad Allahbakhsh. Quality Control in Crowdsourcing: A Survey of Quality Attributes, Assessment Techniques, and Assurance Actions. *ACM Computing Surveys*, 51(1):7:1–7:40, January 2018. ISSN: 0360-0300.

[DS79] Alexander Philip Dawid and Allan M Skene. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1):20–28, 1979. Publisher: Wiley Online Library.

[DSD12] Tom De Smedt and Walter Daelemans. Pattern for python. *The Journal of Machine Learning Research*, 13(1):2063–2067, 2012. Publisher: JMLR. org.

[DCL+21] Dominik Dellermann, Adrian Calma, Nikolaus Lipusch, Thorsten Weber, Sascha Weigel, and Philipp Ebel. The future of human-AI collaboration: a taxonomy of design knowledge for hybrid intelligence systems. *arXiv:2105.03354 [cs]*, May 2021. arXiv: 2105.03354.

[Dem15] Gianluca Demartini. Hybrid human–machine information systems: Challenges and opportunities. *Computer Networks*, 90:5–13, 2015. Publisher: Elsevier.

[DDG+17] Gianluca Demartini, Djellel Eddine Difallah, Ujwal Gadiraju, and Michele Catasta. An Introduction to Hybrid Human-Machine Information Systems. *Foundations and Trends in Web Science*, 7(1):1–87, December 2017. ISSN: 1555-077X.

[DMS20] Gianluca Demartini, Stefano Mizzaro, and Damiano Spina. Human-in-the-loop artificial intelligence for fighting online misinformation: Challenges and opportunities. *The Bulletin of the Technical Committee on Data Engineering*, 43(3):65–74, 2020. Type: Journal article.

[DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. ISSN: 0035-9246. Publisher: [Royal Statistical Society, Wiley].

[DDS+09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255, 2009. tex.organization: IEEE.

[DKS14] John P. Dickerson, Vadim Kagan, and V.S. Subrahmanian. Using sentiment to detect bots on Twitter: Are humans more opinionated than bots? In *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*, pages 620–627, August 2014.

[DFI18] Djellel Difallah, Elena Filatova, and Panos Ipeirotis. Demographics and dynamics of mechanical turk workers. In *Proceedings of the eleventh ACM international conference on web search and data mining*, WSDM '18, pages 135–143, New York, NY, USA. Association for Computing Machinery, 2018. ISBN: 978-1-4503-5581-0. Number of pages: 9 Place: Marina Del Rey, CA, USA.

[DDBA+17] Chris Dijkshoorn, Victor De Boer, Lora Aroyo, and Guus Schreiber. Accurator: nichesourcing for cultural heritage, 2017.

[DFF+19] Alexey Drutsa, Viktoriya Farafonova, Valentina Fedorova, Olga Megorskaya, Evfrosiniya Zerminova, and Olga Zhilinskaya. Practice of Efficient Data Collection via Crowdsourcing at Large-Scale. In arXiv, December 2019. Number: arXiv:1912.04444 arXiv:1912.04444 [cs].

[DK16] PN Druzhkov and VD Kustikova. A survey of deep learning methods and software tools for image classification and object detection. *Pattern Recognition and Image Analysis*, 26(1):9–15, 2016. Publisher: Springer.

[ERC+18] Diego Esteves, Aniketh Janardhan Reddy, Piyush Chawla, and Jens Lehmann. Belittling the source: Trustworthiness indicators to obfuscate fake news on the web. In *Proceedings of the first workshop on fact extraction and VERification (FEVER)*, pages 50–59, 2018.

[Hop] Evaluating resources: the CRAPP TEST, 2015. Type: Electronic article.

[FBC12] Song Feng, Ritwik Banerjee, and Yejin Choi. Syntactic stylometry for deception detection. In *Proceedings of the 50th annual meeting of the association for computational linguistics: Short papers - volume 2*, ACL '12, pages 171–175, Stroudsburg, PA, USA. Association for Computational Linguistics, 2012. Number of pages: 5 Place: Jeju Island, Korea tex.acmid: 2390708.

[FH13] Vanessa Wei Feng and Graeme Hirst. Detecting deceptive opinions with profile compatibility. In *Proceedings of the sixth international joint conference on natural language processing*, pages 338–346, 2013.

[FV16] William Ferreira and Andreas Vlachos. Emergent: a novel data-set for stance classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 1163–1168, 2016.

[FO17] Álvaro Figueira and Luciana Oliveira. The current state of fake news: challenges and opportunities. *Procedia Computer Science*, 121:817–825, 2017. Publisher: Elsevier.

[FKT+13] Ailbhe Finnerty, Pavel Kucherbaev, Stefano Tranquillini, and Gregorio Convertino. Keep it simple: Reward and task design in crowdsourcing. In *Proceedings of the biannual conference of the italian chapter of SIGCHI*, CHItaly '13, 14:1–14:4, New York, NY, USA. ACM, 2013. ISBN: 978-1-4503-2061-0. Number of pages: 4 Place: Trento, Italy tex.acmid: 2499168 tex.articleno: 14.

[Fur16] Marco Furini. On gamifying the transcription of digital video lectures. en. *Entertainment Computing*, 14:23–31, May 2016. ISSN: 1875-9521.

[GYB17] Ujwal Gadiraju, Jie Yang, and Alessandro Bozzon. Clarity is a Worthwhile Quality: On the Role of Task Clarity in Microtask Crowdsourcing. In *Proceedings of the 28th ACM Conference on Hypertext and Social Media*, HT '17, pages 5–14, New York, NY, USA. Association for Computing Machinery, July 2017. ISBN: 978-1-4503-4708-2.

[GMJM+16] H. Garcia-Molina, M. Joglekar, A. Marcus, A. Parameswaran, and V. Verroios. Challenges in data crowdsourcing. *IEEE Trans. on Knowledge and Data Engineering*, 2016.

[Gar00] Howard E. Gardner. *Intelligence Reframed: Multiple Intelligences for the 21st Century*. en. Hachette UK, September 2000. ISBN: 978-0-465-01314-2. Google-Books-ID: Qkw4DgAAQBAJ.

[GRH+20] Ralph Gasser, Luca Rossetto, Silvan Heller, and Heiko Schuldt. Cottontail DB: An open source database system for multimedia retrieval and analysis. In *Proceedings of the 28 [U+1D57]^h ACM international conference on multimedia*. ACM, 2020.

[GMM+15] Yashesh Gaur, Florian Metze, Yajie Miao, and Jeffrey P. Bigham. Using keyword spotting to help humans correct captioning faster. en. In *Interspeech 2015*, pages 2829–2833. ISCA, September 2015.

[GGM+18] Antonio Ghezzi, Donata Gabelloni, Antonella Martini, and Angelo Natalicchio. Crowdsourcing: A Review and Suggestions for Future Research. en. *International Journal of Management Reviews*, 20(2):343–363, 2018. ISSN: 1468-2370. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/ijmr.12135.

[GDD+15] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Region-based convolutional networks for accurate object detection and segmentation. *IEEE tx on pattern analysis and machine intelligence*, 38(1):142–158, 2015. Publisher: IEEE.

[GMA+18] Jennifer Golbeck, Matthew Mauriello, Brooke Auxier, Keval H. Bhanushali, Christopher Bonk, Mohamed Amine Bouzaghrane, Cody Buntain, Riya Chanduka, Paul Cheakalos, Jennine B. Everett, Waleed Falak, Carl Gieringer, Jack Graney, Kelly M. Hoffman, Lindsay Huth, Zhenya Ma, Mayanka Jha, Misbah Khan, Varsha Kori, Elo Lewis, George Mirano, William T. Mohn IV, Sean Mussenden, Tammie M. Nelson, Sean Mcwillie, Akshat

Pant, Priya Shetye, Rusha Shrestha, Alexandra Steinheimer, Aditya Subramanian, and Gina Visnansky. Fake news vs satire: A dataset and analysis. In *Proceedings of the 10th ACM conference on web science*, WebSci '18, pages 17–21, New York, NY, USA. ACM, 2018. ISBN: 978-1-4503-5563-6. Number of pages: 5 Place: Amsterdam, Netherlands tex.acmid: 3201100.

[Gra18] Lucas Graves. Understanding the promise and limits of automated fact-checking. Report, Reuters Institute for the Study of Journalism et University of Oxford, 2018.

[GHEG19] Nina Grgić-Hlača, Christoph Engel, and Krishna P. Gummadi. Human Decision Making with Machine Assistance: An Experiment on Bailing and Jailing. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW):178:1–178:25, November 2019.

[GPGM12] Stephen Guo, Aditya Parameswaran, and Hector Garcia-Molina. So who won? dynamic max discovery with the crowd. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 385–396, New York, NY, USA. Association for Computing Machinery, May 2012. ISBN: 978-1-4503-1247-9.

[GKC+14] Aditi Gupta, Ponnurangam Kumaraguru, Carlos Castillo, and Patrick Meier. Tweetcred: Real-time credibility assessment of content on twitter. In *International conference on social informatics*, pages 228–243, 2014. tex.organization: Springer.

[GGN+08] Isabelle Guyon, Steve Gunn, Masoud Nikravesh, and Lofti A. Zadeh. *Feature Extraction: Foundations and Applications*. en. Springer, November 2008. ISBN: 978-3-540-35488-8. Google-Books-ID: FOTzBwAAQBAJ.

[HPR19] Louisa Ha, Loarre Andreu Perez, and Rik Ray. Mapping recent development in scholarship on fake news and misinformation, 2008-2017: Disciplinary contribution, topics, and impact. *American Behavioral Scientist*, August:1–26, 2019. Type: Journal article.

[Han10] Eric Hand. People power: networks of human minds are taking citizen science to a new level. *Nature*, 466(7307):685–688, 2010. Publisher: Nature Publishing Group.

[Har14] Jay R Harman. *Collateral damage: The imperiled status of truth in american public discourse and why it matters to you*. Author House, Bloomington, Indiana, 2014. ISBN: 978-1-4918-5572. Type: Book.

[Hol10] Rose Holley. Crowdsourcing: How and why should libraries do it? *D-Lib magazine*, 16(3/4 Ma), 2010. Publisher: Corporation for National Research Initiative (CNRI).

[Hol16] Andreas Holzinger. Interactive machine learning for health informatics: when do we need the human-in-the-loop? en. *Brain Informatics*, 3(2):119–131, June 2016. ISSN: 2198-4026.

[HZC+17] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications, 2017.

[How06] Jeff Howe. The Rise of Crowdsourcing. en. (14):5, 2006.

[HRB14] Chang Hu, Philip Resnik, and Benjamin B. Bederson. Crowdsourced Monolingual Translation. *ACM Transactions on Computer-Human Interaction*, 21(4):22:1–22:35, August 2014. ISSN: 1073-0516.

[HWQ21] Gaoping Huang, Meng-Han Wu, and Alexander J. Quinn. Task design for crowdsourcing complex cognitive skills. In *Extended abstracts of the 2021 CHI conference on human factors in computing systems*. Association for Computing Machinery, New York, NY, USA, 2021. ISBN: 978-1-4503-8095-9. Number of pages: 7 tex.articleno: 51.

[Ipe10] Panagiotis G Ipeirotis. Analyzing the amazon mechanical turk marketplace. *XRDS: Crossroads, The ACM Magazine for Students*, 17(2):16–21, 2010. Publisher: ACM New York, NY, USA.

[KKMF12] Gabriella Kazai, Jaap Kamps, and Natasa Milic-Frayling. The face of quality in crowdsourcing relevance labels: Demographics, personality and labeling accuracy. In *Proceedings of the 21st ACM international conference on information and knowledge management*, CIKM '12, Maui, Hawaii, USA, 2012.

[KT18] John D. Kelleher and Brendan Tierney. *Data Science*. en. MIT Press, April 2018. ISBN: 978-0-262-34703-7. Google-Books-ID: UlpVDwAAQBAJ.

[KSS21] Gavin Kerrigan, Padhraic Smyth, and Mark Steyvers. Combining Human Predictions with Model Probabilities via Confusion Matrices and Calibration. In *Advances in Neural Information Processing Systems*, volume 34, pages 4421–4434. Curran Associates, Inc., 2021.

[KJR+20] Omar Shahbaz Khan, Björn Þór Jónsson, Stevan Rudinac, Jan Zahálka, Hanna Ragnarsdóttir, Þórhildur Þorleiksdóttir, Gylfi Þór Guðmundsson, Laurent Amsaleg, and Marcel Worring. Interactive Learning for Multimedia at Large. en. In Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins, editors, *Advances in Information Retrieval*, Lecture Notes in Computer Science, pages 495–510, Cham. Springer International Publishing, 2020. ISBN: 978-3-030-45439-5.

[KPB+20] Lev Konstantinovskiy, Oliver Price, Mevan Babakar, and Arkaitz Zubiaga. Towards automated factchecking: Developing an annotation schema and benchmark for consistent automated claim detection, 2020. tex.archiveprefix: arXiv.

[KJMH+19] Kamran Kowsari, Kiana Jafari Meimandi, Mojtaba Heidarysafa, Sanjana Mendu, Laura Barnes, and Donald Brown. Text Classification Algorithms: A Survey. en. *Information*, 10(4):150, April 2019. Number: 4 Publisher: Multidisciplinary Digital Publishing Institute.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[KJ13] Max Kuhn and Kjell Johnson. *Applied Predictive Modeling*. en. Springer New York, New York, NY, 2013. ISBN: 978-1-4614-6848-6 978-1-4614-6849-3.

[KLW+03] Kuncheva, L.I., C.J. Whitaker, C.A. Shipp, and R.P.W. Duin. Limits on the majority vote accuracy in classifier fusion. *Pattern Analysis and Applications*, 2003.

[LUT+17] Brenden M. Lake, Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel J. Gershman. Building machines that learn and think like

people. en. *Behavioral and Brain Sciences*, 40, 2017. ISSN: 0140-525X, 1469-1825. Publisher: Cambridge University Press.

[LMN+17] Walter S. Lasecki, Christopher D. Miller, Iftekhar Naim, Raja Kushalnagar, Adam Sadilek, Daniel Gildea, and Jeffrey P. Bigham. Scribe: deep integration of human and machine intelligence to caption speech in real time. *Communications of the ACM*, 60(9):93–100, August 2017. ISSN: 0001-0782.

[LBB+18] David MJ Lazer, Matthew A Baum, Yochai Benkler, Adam J Berinsky, Kelly M Greenhill, Filippo Menczer, Miriam J Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, and others. The science of fake news. *Science (New York, N.Y.)*, 2018. Publisher: American Association for the Advancement of Science.

[LY12] Matthew Lease and Emine Yilmaz. Crowdsourcing for information retrieval. *ACM SIGIR Forum*, 45(2):66–75, January 2012. ISSN: 0163-5840.

[LBD+89] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4):541–551, December 1989. ISSN: 0899-7667. Conference Name: Neural Computation.

[LBB+98] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998. ISSN: 1558-2256. Conference Name: Proceedings of the IEEE.

[LS20] In Lee and Yong Jae Shin. Machine learning for enterprises: Applications, algorithm selection, and challenges. en. *Business Horizons*. ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING, 63(2):157–170, March 2020. ISSN: 0007-6813.

[LY14] H. Li and B. Yu. Error rate bounds and iterative weighted majority voting for crowdsourcing. *ArXiv e-prints*, 2014. arXiv: 1411.4086.

[LGS+20] Yunyao Li, Tyrone Grandison, Patricia Silveyra, Ali Douraghy, Xinyu Guan, Thomas Kieselbach, Chengkai Li, and Haiqi Zhang. Jennifer for COVID-19: An NLP-Powered chatbot built for the people and by the people to combat misinformation. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, 2020.

[Liu12] Bing Liu. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167, 2012. Publisher: Morgan & Claypool Publishers.

[LLO+12] Xuan Liu, Meiyu Lu, Beng Chin Ooi, Yanyan Shen, Sai Wu, and Meihui Zhang. CDAS: A crowdsourcing data analytics system. *Proceedings of the VLDB Endowment*, 2012. Publisher: VLDB Endowment.

[LMLM+17] Jose Llamas, Pedro M Lerones, Roberto Medina, Eduardo Zalama, and Jaime Gómez-García-Bermejo. Classification of architectural heritage images using deep learning techniques. *Applied Sciences*, 7(10):992, 2017. Publisher: Multidisciplinary Digital Publishing Institute.

[LM14] C. Lofi and K. E. Maarry. Design patterns for hybrid algorithmic-crowdsourcing workflows. In *2014 IEEE 16th conference on business informatics*, 2014.

[MGB15] Kinda El Maarry, Ulrich Güntzer, and Wolf-Tilo Balke. Realizing Impact Sourcing by Adaptive Gold Questions: A Socially Responsible Measure for Workers' Trustworthiness. en. In Xin Luna Dong, Xiaohui Yu, Jian Li, and Yizhou Sun, editors, *Web-Age Information Management*, Lecture Notes in Computer Science, pages 17–29, Cham. Springer International Publishing, 2015. ISBN: 978-3-319-21042-1.

[MKM+19] Prathmesh Madhu, Ronak Kosti, Lara Mührenberg, Peter Bell, Andreas Maier, and Vincent Christlein. Recognizing characters in art history using deep learning. In *Proceedings of the 1st workshop on structuring and understanding of multimedia HeritAge contents*, SUMAC '19, pages 15–22, New York, NY, USA. Association for Computing Machinery, 2019. ISBN: 978-1-4503-6910-7. Number of pages: 8 Place: Nice, France.

[MRS08] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. en. Cambridge University Press, July 2008. ISBN: 978-0-521-86571-5. Google-Books-ID: GNvtngEACAAJ.

[MCH+17] Ke Mao, Licia Capra, Mark Harman, and Yue Jia. A survey of the use of crowdsourcing in software engineering. en. *Journal of Systems and Software*, 126:57–84, April 2017. ISSN: 0164-1212.

[MP15] Adam Marcus and Aditya Parameswaran. Crowdsourced Data Management: Industry and Academic Perspectives. *Foundations and Trends in Databases*, 6(1-2):1–161, December 2015. ISSN: 1931-7883.

[MH14] David M Markowitz and Jeffrey T Hancock. Linguistic traces of a scientific fraud: The case of Diederik Stapel. *PloS one*, 9(8):e105937, 2014. Publisher: Public Library of Science.

[McH12] Mary L McHugh. Interrater reliability: the kappa statistic. *Biochemia medica: Biochemia medica*, 22(3):276–282, 2012. Publisher: Medicinska naklada.

[MJGS+11] Pablo N Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. DBpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems*, pages 1–8, 2011.

[MS09] Rada Mihalcea and Carlo Strapparava. The Lie Detector: Explorations in the Automatic Recognition of Deceptive Language. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 309–312, Suntec, Singapore. Association for Computational Linguistics, August 2009.

[MCK+13] Luyi Mo, Reynold Cheng, Ben Kao, Xuan S. Yang, Chenghui Ren, Siyu Lei, David W. Cheung, and Eric Lo. Optimizing plurality for human intelligence tasks. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, CIKM '13, pages 1929–1938, New York, NY, USA. Association for Computing Machinery, October 2013. ISBN: 978-1-4503-2263-8.

[MRT18] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

[MHK16] B. Morschheuser, J. Hamari, and J. Koivisto. Gamification in crowdsourcing: A review. In *49th hawaii international conference on system sciences (HICSS)*, 2016.

[MTD16] Evangelos Mourelatos, Manolis Tzagarakis, and Efthalia Dimara. A review of online crowdsourcing platforms. en. *South-Eastern Europe Journal of Economics*, 14(1), 2016. Number: 1.

[NBG+15] Kawa Nazemi, Dirk Burkhardt, Egils Ginters, and Jorn Kohlhammer. Semantics Visualization–Definition, approaches and challenges. *Procedia Computer Science*:75–83, 2015. Publisher: Elsevier.

[NK20] Dorit Nevo and Julia Kotlarsky. Crowdsourcing as a strategic IS sourcing phenomenon: Critical review and insights for future research. en. *The Journal of Strategic Information Systems*. 2020 Review Issue, 29(4):101593, December 2020. ISSN: 0963-8687.

[NKK+18] An T Nguyen, Aditya Kharosekar, Saumyaa Krishnan, Siddhesh Krishnan, Elizabeth Tate, Bryon C Wallace, and Matthew Lease. Believe it or not: Designing a human-ai partnership for mixed-initiative fact-checking. In *UIST '18: The 31st annual ACM symposium on user interface software and technology*, pages 189–199, 2018.

[NKL+18] An Thanh Nguyen, Aditya Kharosekar, Matthew Lease, and Byron C Wallace. An interpretable joint graphical model for fact-checking from crowds. In *AAAI conference on artificial intelligence*, pages 1511–1518, 2018.

[NVL+14] Evangelos Niforatos, Athanasios Vourvopoulos, Marc Langheinrich, Pedro Campos, and Andre Doria. Atmos: a hybrid crowdsourcing approach to weather estimation. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, UbiComp '14 Adjunct, pages 135–138, New York, NY, USA. Association for Computing Machinery, September 2014. ISBN: 978-1-4503-3047-3.

[NAM21] Curtis G. Northcutt, Anish Athalye, and Jonas Mueller. Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks, November 2021. Number: arXiv:2103.14749 arXiv:2103.14749 [cs, stat].

[NOC+14] Archana Nottamkandath, Jasper Oosterman, Davide Ceolin, Wan Fokkink, and others. Automated evaluation of crowdsourced annotations in the cultural heritage domain. In *URSW*, pages 25–36, 2014.

[OWK99] Margaret T. O'Hara, Richard T. Watson, and C. Bruce Kavan. Managing the three Levels of Change. *Information Systems Management*, 16(3):63–70, June 1999. ISSN: 1058-0530. Publisher: Taylor & Francis _eprint: https://doi.org/10.1201/1078/43197.16.3.19990601/31317.9.

[OSL+11] David Oleson, Alexander Sorokin, Greg Laughlin, Vaughn Hester, John Le, and Lukas Biewald. Programmatic gold: Targeted and scalable quality assurance in crowdsourcing. In *Workshops at the twenty-fifth AAAI conference on artificial intelligence*, 2011.

[OSS16] Alex C Olivieri, Roland Schegg, and Maria Sokhn. Cityzen: a social platform for cultural heritage focused tourism. In *Proceedings of the 8th international conference on management of digital EcoSystems*, pages 129–136, 2016.

[OA11] Johan Oomen and Lora Aroyo. Crowdsourcing in the cultural heritage domain: opportunities and challenges. In *Proceedings of the 5th international conference on communities and technologies*, pages 138–149, 2011.

[OND+14] Jasper Oosterman, Archana Nottamkandath, Chris Dijkshoorn, Alessandro Bozzon, Geert-Jan Houben, and Lora Aroyo. Crowdsourcing knowledge-intensive tasks in cultural heritage. In *Proceedings of the 2014 ACM conference on web science*, WebSci '14, pages 267–268, New York, NY, USA. Association for Computing Machinery, 2014. ISBN: 978-1-4503-2622-3. Number of pages: 2 Place: Bloomington, Indiana, USA.

[ONB+17] Sergio Oramas, Oriol Nieto, Francesco Barbieri, and Xavier Serra. Multi-label music genre classification from audio, text and images using deep features. In *Proceedings of the 18th international society for music information retrieval conference*, pages 23–30, Suzhou, China. ISMIR, October 2017. tex.venue: Suzhou, China.

[OCC+11] Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T. Hancock. Finding Deceptive Opinion Spam by Any Stretch of the Imagination. *arXiv:1107.4557 [cs]*, July 2011. arXiv: 1107.4557.

[PY09] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009. Publisher: IEEE.

[PF14] Dimitris Paraschakis and Marie Gustafsson Friberger. Playful crowdsourcing of archival metadata through social networks, 2014.

[PVG+11] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and others. Scikit-learn: Machine learning in python. *Journal of machine learning research*:2825–2830, 2011.

[PBJ+15] James W. Pennebaker, Ryan L. Boyd, Kayla Jordan, and Kate Blackburn. The Development and Psychometric Properties of LIWC2015. eng, September 2015. Accepted: 2015-09-16T13:00:41Z.

[PFB01] James W Pennebaker, Martha E Francis, and Roger J Booth. Linguistic inquiry and word count: LIWC 2001. *Mahway: Lawrence Erlbaum Associates*, 71(2001), 2001.

[PSM14] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[Per18] Kleiner Perkins. KPCB - Internet Trends 2018. en, 2018.

[Pis17] Dina Pisarevskaya. Deception detection in news reports in the russian language: Lexics and discourse. In *Proceedings of the 2017 EMNLP workshop: Natural language processing meets journalism*, pages 74–79, 2017.

[PNV+02] Kostas Proedrou, Ilia Nouretdinov, Volodya Vovk, and Alex Gammerman. Transductive confidence machines for pattern recognition. In *European conference on machine learning*, pages 381–390, 2002. tex.organization: Springer.

[PEOOAP21] Marta Pérez-Escolar, Eva Ordóñez-Olmedo, and Purificación Alcaide-Pulido. Fact-Checking Skills And Project-Based Learning About Infodemic And Disinformation. en. *Thinking Skills and Creativity*, 41:100887, September 2021. ISSN: 1871-1871.

[PRKL+18] Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea. Automatic detection of fake news. In *Proceedings of the 27th international conference on computational linguistics*, 2018.

[QB11] Alexander J. Quinn and Benjamin B. Bederson. Human computation: a survey and taxonomy of a growing field. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1403–1412, New York, NY, USA. Association for Computing Machinery, May 2011. ISBN: 978-1-4503-0228-9.

[QVHTT+13] Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Lam Ngoc Tran, and Karl Aberer. An evaluation of aggregation techniques in crowdsourcing. In *Web information systems engineering – WISE 2013*, pages 1–15, 2013.

[RBG+09] M Jordan Raddick, Georgia Bracey, Pamela L Gay, Chris J Lintott, Phil Murray, Kevin Schawinski, Alexander S Szalay, and Jan Vandenberg. Galaxy zoo: Exploring the motivations of citizen science volunteers. *arXiv preprint arXiv:0909.2925*, 2009.

[RYZ+10] Vikas C. Raykar, Shipeng Yu, Linda H. Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. Learning From Crowds. *Journal of Machine Learning Research*, 11(43):1297–1322, 2010. ISSN: 1533-7928.

[Reh17] Georg Rehm. An infrastructure for empowering internet users to handle fake news and other online media phenomena. In *International conference of the german society for computational linguistics and language technology*, pages 216–231, 2017. tex.organization: Springer.

[RZZ+15] Ju Ren, Yaoxue Zhang, Kuan Zhang, and Xuemin Shen. Exploiting mobile crowdsourcing for pervasive cloud services: challenges and solutions. *IEEE Communications Magazine*, 53(3):98–105, 2015. ISSN: 0163-6804.

[ROFC+19] Benjamin Renoust, Matheus Oliveira Franca, Jacob Chan, Noa Garcia, Van Le, Ayaka Uesaka, Yuta Nakashima, Hajime Nagahara, Jueren Wang, and Yutaka Fujioka. Historical and modern features for buddha statue classification. In *Proceedings of the 1st workshop on structuring and understanding of multimedia HeritAge contents*, SUMAC '19, pages 23–30, New York, NY, USA. Association for Computing Machinery, 2019. ISBN: 978-1-4503-6910-7. Number of pages: 8 Place: Nice, France.

[RHCM20] Laura Rettig, Regula Hänggli, and Philippe Cudré-Mauroux. The best of both worlds: Context-powered word embedding combinations for longitudinal text analysis. In *Proceedings of the IEEE international conference on big data*, 2020.

[RSS+21] Laura Rettig, Shaban Shabani, Loris Sauter, Philippe Cudré-Mauroux, Maria Sokhn, and Heiko Schuldt. City-stories: Combining entity linking, multimedia retrieval, and crowdsourcing to make historical data accessible. In *In the proceedings of the inetrnational conference on web engineering*. Springer International Publishing, 2021. ISBN: 978-3-030-74296-6.

[RSG16] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 1135–1144, New York, NY, USA. Association for Computing Machinery, August 2016. ISBN: 978-1-4503-4232-2.

[RSP+20] Kevin Roitero, Michael Soprano, Beatrice Portelli, Damiano Spina, Vincenzo Della Mea, Giuseppe Serra, Stefano Mizzaro, and Gianluca Demartini. The covid-19 infodemic: Can the crowd judge recent misinformation objectively? In *Proceedings of the 29th ACM international conference on information & knowledge management*, 2020.

[RGS14] Luca Rossetto, Ivan Giangreco, and Heiko Schuldt. Cineast: a multi-feature sketch-based video retrieval engine. In *2014 IEEE international symposium on multimedia*, pages 18–23, 2014. tex.organization: IEEE.

[RGT+16] Luca Rossetto, Ivan Giangreco, Claudiu Tanase, and Heiko Schuldt. Vitrivr: A flexible retrieval stack supporting multiple query modes for searching in multimedia collections. In *Proceedings of the 24th ACM international conference on Multimedia*, pages 1183–1186, 2016.

[Rot03] David J Rothkopf. When the buzz bites back. *The Washington Post*, (May 11th):BO1, 2003. Type: Magazine article.

[RCC15] Victoria L. Rubin, Yimin Chen, and Niall J. Conroy. Deception detection for news: Three types of fakes. In *Proceedings of the 78th ASIS&T annual meeting: Information science with impact: Research*

in and for the community, ASIST '15, 83:1–83:4, Silver Springs, MD, USA. American Society for Information Science, 2015. ISBN: 0-87715-547-X. Number of pages: 4 Place: St. Louis, Missouri tex.acmid: 2857153 tex.articleno: 83.

[RCC+16] Victoria L Rubin, Niall Conroy, Yimin Chen, and Sarah Cornwell. Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the second workshop on computational approaches to deception detection*, pages 7–17, 2016.

[RSL17] Natali Ruchansky, Sungyong Seo, and Yan Liu. Csi: A hybrid deep model for fake news detection. In *Proceedings of the 2017 ACM on conference on information and knowledge management*, pages 797–806, 2017.

[RN09] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. English. Pearson, Upper Saddle River, 3rd edition edition, December 2009. ISBN: 978-0-13-604259-4.

[SZD+17] Mehrnoosh Sameki, Tianyi Zhang, Linli Ding, Margrit Betke, and Danna Gurari. Crowd-o-meter: Predicting if a person is vulnerable to believe political claims. In *Fifth AAAI conference on human computation and crowdsourcing*, 2017.

[Sar21] Iqbal H. Sarker. Machine Learning: Algorithms, Real-World Applications and Research Directions. en. *SN Computer Science*, 2(3):160, March 2021. ISSN: 2661-8907.

[SSD12] Daniel Schall, Florian Skopik, and Schahram Dustdar. Expert Discovery and Interactions in Mixed Service-Oriented Systems. *IEEE Transactions on Services Computing*, 5(2):233–245, April 2012. ISSN: 1939-1374. Conference Name: IEEE Transactions on Services Computing.

[SR09] Satoshi Sekine and Elisabete Ranchhod. *Named Entities: Recognition, Classification, and Use*. en. John Benjamins Publishing, 2009. ISBN: 978-90-272-2249-7. Google-Books-ID: rIM_9TWOmNoC.

[SCS+21] Shaban Shabani, Zarina Charlesworth, Maria Sokhn, and Heiko Schuldt. SAMS: Human-in-the-loop approach to combat the sharing of digital misinformation. In *Proceedings of the AAAI 2021 spring symposium on combining machine learning and knowledge engineering (AAAI-MAKE 2021)*, 2021.

[SLS18] Shaban Shabani, Zhan Liu, and Maria Sokhn. Semantic network visualization of cultural heritage data. In *International conference on web engineering*, pages 288–291, 2018. tex.organization: Springer.

[SS17] Shaban Shabani and Maria Sokhn. Gaming as a gateway: Ensuring quality control for crowdsourced data. In *In proceedings of international conference on cooperative design, visualization, and engineering (CDVE)*. Springer International Publishing, 2017. ISBN: 978-3-319-66805-5.

[SS18] Shaban Shabani and Maria Sokhn. Hybrid machine-crowd approach for fake news detection. In *2018 IEEE 4th international conference on collaboration and internet computing (CIC)*, pages 299–306, 2018. tex.organization: IEEE.

[SSR+17] Shaban Shabani, Maria Sokhn, Laura Rettig, Philippe Cudré-Mauroux, Lukas Beck, Claudiu Tanase, and Heiko Schuldt. Citystories: A multimedia hybrid content and entity retrieval system for historical data. In *HistoInformatics@CIKM*, pages 22–29, 2017.

[SSS20] Shaban Shabani, Maria Sokhn, and Heiko Schuldt. Hybrid human-machine classification system for cultural heritage data. In *Proceedings of the 2nd workshop on structuring and understanding of multimedia HeritAge contents*, SUMAC'20, pages 49–56, Seattle, WA, USA, 2020. Number of pages: 8.

[SSW+17] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. *SIGKDD Explor. Newsl.*, 19, September 2017. Number of pages: 15 Place: New York, NY, USA Publisher: Association for Computing Machinery tex.issue_date: June 2017.

[SZ14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[SSS17] Jaspreet Singh, Gurvinder Singh, and Rajinder Singh. Optimization of sentiment analysis using machine learning classifiers. *Human-centric Computing and information Sciences*, 7(1):1–12, 2017. Publisher: Springer.

[SD12] Tom De Smedt and Walter Daelemans. Pattern for python. *Journal of Machine Learning Research*, 13(Jun):2063–2067, 2012.

[SOJ+08] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Ng. Cheap and fast – but is it good? Evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 conference on empirical methods in natural language processing*, pages 254–263, Honolulu, Hawaii. Association for Computational Linguistics, October 2008.

[Spr20] Marianna Spring. Coronavirus: The seven types of people who start and spread viral misinformation. *BBC Trending*, May 2020. Publisher: BBC.

[SR17] Ralph Stair and George Reynolds. *Fundamentals of Information Systems*. en. Cengage Learning, March 2017. ISBN: 978-1-337-51563-4. Google-Books-ID: GtVBDgAAQBAJ.

[SBD+21] Stefan Studer, Thanh Binh Bui, Christian Drescher, Alexander Hanuschkin, Ludwig Winkler, Steven Peters, and Klaus-Robert Müller. Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology. en. *Machine Learning and Knowledge Extraction*, 3(2):392–413, June 2021. ISSN: 2504-4990. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

[SSS+17] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852, 2017.

[SRY+14] Chong Sun, Narasimhan Rampalli, Frank Yang, and AnHai Doan. Chimera: Large-scale classification using machine learning, rules, and crowdsourcing. *Proceedings of the VLDB Endowment*, 2014. Publisher: VLDB Endowment.

[SB98] R.S. Sutton and A.G. Barto. Reinforcement Learning: An Introduction. *IEEE Transactions on Neural Networks*, 9(5):1054–1054, September 1998. ISSN: 1941-0093. Conference Name: IEEE Transactions on Neural Networks.

[SIV+17] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

[Sze10] Richard Szeliski. *Computer Vision: Algorithms and Applications*. en. Springer Science & Business Media, September 2010. ISBN: 978-1-84882-935-0. Google-Books-ID: bXzAlkODwa8C.

[TP10] Yla R. Tausczik and James W. Pennebaker. The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods. en. *Journal of Language and Social Psychology*, 29(1):24–54, March 2010. ISSN: 0261-927X. Publisher: SAGE Publications Inc.

[TRP21] Marina Tavra, Ivan Racetin, and Josip Peroš. The role of crowdsourcing and social media in crisis mapping: a case study of a wildfire reaching Croatian City of Split. *Geoenvironmental Disasters*, 8(1):10, April 2021. ISSN: 2197-8670.

[TCA+17] Roselyne B. Tchoua, Kyle Chard, Debra J. Audus, Logan T. Ward, Joshua Lequieu, Juan J. De Pablo, and Ian T. Foster. Towards a Hybrid Human-Computer Scientific Information Extraction Pipeline. In *2017 IEEE 13th International Conference on e-Science (e-Science)*, pages 109–118, October 2017.

[TVC+18] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 conference of the north american chapter of the association for computational linguistics*, pages 809–819, 2018.

[TCZ+18] Yongxin Tong, Lei Chen, Zimu Zhou, H. V. Jagadish, Lidan Shou, and Weifeng Lv. SLADE: A Smart Large-Scale Task Decomposer in Crowdsourcing. *IEEE Transactions on Knowledge and Data Engineering*, 30(8):1588–1601, August 2018. ISSN: 1558-2191. Conference Name: IEEE Transactions on Knowledge and Data Engineering.

[Tra09] Jennifer Trant. Tagging, folksonomy and art museums: Early experiments and ongoing research, 2009.

[TOH+14] Myriam C Traub, Jacco van Ossenbruggen, Jiyin He, and Lynda Hardman. Measuring the effectiveness of gamesourcing expert oil painting annotations. In *European conference on information retrieval*, pages 112–123, 2014. tex.organization: Springer.

[TSGR+18] Sebastian Tschiatschek, Adish Singla, Manuel Gomez Rodriguez, Arpit Merchant, and Andreas Krause. Fake news detection in social networks via crowd signals. In *The web conference 2018*, WWW '18, pages 517–524, Lyon, France, 2018.

[TSM14] Piyoros Tungthamthiti, Kiyoaki Shirai, and Masnizah Mohd. Recognition of sarcasms in tweets based on concept level sentiment analysis and supervised learning approaches. In *Proceedings of the 28th Pacific Asia conference on language, information and computing*, pages 404–413, 2014.

[TL18] Nicol Turner Lee. Detecting racial bias in algorithms and machine learning. *Journal of Information, Communication and Ethics in Society*, 16(3):252–260, January 2018. ISSN: 1477-996X. Publisher: Emerald Publishing Limited.

[UG14] Alper Kursat Uysal and Serkan Gunal. The impact of preprocessing on text classification. en. *Information Processing & Management*, 50(1):104–112, January 2014. ISSN: 0306-4573.

[VBD14] Norases Vesdapunt, Kedar Bellare, and Nilesh Dalvi. Crowdsourcing algorithms for entity resolution. *Proceedings of the VLDB Endowment*, 7(12):1071–1082, August 2014. ISSN: 2150-8097.

[VC12] G Vinodhini and RM Chandrasekaran. Sentiment analysis and opinion mining: a survey. *International Journal*, 2(6):282–292, 2012.

[VSJ+17] Svitlana Volkova, Kyle Shaffer, Jin Yea Jang, and Nathan Hodas. Separating Facts from Fiction: Linguistic Models to Classify Suspicious and Trusted News Posts on Twitter. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 647–653, Vancouver, Canada. Association for Computational Linguistics, July 2017.

[vAh05] Luis von Ahn. *Human computation*. PhD thesis, CMU, Pittsburgh, PA, USA, 2005.

[vAh13] Luis von Ahn. Duolingo: Learn a language for free while helping to translate the web. In *Proc. of the international conference on intelligent user interfaces*. ACM, 2013.

[AD04a] Luis von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 2004.

[AD04b] Luis von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 319–326, New York, NY, USA. Association for Computing Machinery, April 2004. ISBN: 978-1-58113-702-6.

[AD08] Luis von Ahn and Laura Dabbish. Designing games with a purpose. *Communications of The Acm*, 2008. Publisher: ACM.

[ALB06] Luis von Ahn, Ruoran Liu, and Manuel Blum. Peekaboom: a game for locating objects in images. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 55–64, New York, NY, USA. Association for Computing Machinery, April 2006. ISBN: 978-1-59593-372-0.

[VAMM+08] Luis Von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. Recaptcha: Human-based character recognition via web security measures. *Science (New York, N.Y.)*, 321(5895):1465–1468, 2008. Publisher: American Association for the Advancement of Science.

[Vov02] Vladimir Vovk. On-line confidence machines are well-calibrated. In *Proceedings of the 43rd symposium on foundations of computer science*, 2002.

[WNSM+17] Byron C Wallace, Anna Noel-Storr, Iain J Marshall, Aaron M Cohen, Neil R Smalheiser, and James Thomas. Identifying reports of randomized controlled trials (RCTs) via a hybrid machine learning and crowdsourcing approach. *Journal of the American Medical Informatics Association*, 24(6):1165–1168, November 2017. ISSN: 1067-5027.

[WWW+18] Jiangtao Wang, Leye Wang, Yasha Wang, Daqing Zhang, and Linghe Kong. Task Allocation in Mobile Crowd Sensing: State-of-the-Art and Future Opportunities. *IEEE Internet of Things Journal*, 5(5):3747–3757, October 2018. ISSN: 2327-4662. Conference Name: IEEE Internet of Things Journal.

[Wan17] William Yang Wang. "Liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th annual meeting of the association for computational linguistics*, 2017.

[WKW16] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):1–40, 2016. Publisher: SpringerOpen.

[WC11] Andrea Wiggins and Kevin Crowston. From conservation to crowdsourcing: A typology of citizen science. In *2011 44th Hawaii international conference on system sciences*, pages 1–10, 2011. tex.organization: IEEE.

[WH00] R. Wirth and Jochen Hipp. Crisp-dm: towards a standard process modell for data mining. In 2000.

[WSWA+19] Megan A. Witherow, Cem Sazara, Irina M. Winter-Arboleda, Mohamed I. Elbakary, Mecit Cetin, and Khan M. Iftekharuddin. Floodwater detection on roadways from crowdsourced images. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 7(5-6):529–540, November 2019. ISSN: 2168-1163. Publisher: Taylor & Francis.

[ZSB+19] Savvas Zannettou, Michael Sirivianos, Jeremy Blackburn, and Nicolas Kourtellis. The web of false information: Rumors, fake news, hoaxes, clickbait , and various other shenanigans. *Journal of Data and Information Quality*, 11(3):37, 2019. Type: Journal article.

[ZRM+18] Amy X Zhang, Aditya Ranganathan, Sarah Emlen Metz, Scott Appling, Connie Moon Sehat, Norman Gilmore, Nick B Adams, Emmanuel Vincent, Jennifer Lee, Martin Robbins, and others. A structured response to misinformation: Defining and annotating credibility indicators in news articles. In *The web conference 2018*, pages 603–612, 2018.

[ZZL+19] Daniel Zhang, Yang Zhang, Qi Li, Thomas Plummer, and Dong Wang. CrowdLearn: A Crowd-AI Hybrid System for Deep Learning-based Damage Assessment Applications. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pages 1221–1232, July 2019. ISSN: 2575-8411.

[ZYL+19] Xinglin Zhang, Zheng Yang, Yunhao Liu, and Shaohua Tang. On Reliable Task Assignment for Spatial Crowdsourcing. *IEEE Transactions on Emerging Topics in Computing*, 7(1):174–186, January 2019. ISSN: 2168-6750. Conference Name: IEEE Transactions on Emerging Topics in Computing.

[ZFW+17] Bo Zhao, Jiashi Feng, Xiao Wu, and Shuicheng Yan. A survey on deep learning-based fine-grained object classification and semantic segmentation. *International Journal of Automation and Computing*, 14(2):119–135, 2017. Publisher: Springer.

[ZGS+10] Matthew Zook, Mark Graham, Taylor Shelton, and Sean Gorman. Volunteered Geographic Information and Crowdsourcing Disaster Relief: A Case Study of the Haitian Earthquake. en. *World Medical & Health Policy*, 2(2):7–33, 2010. ISSN: 1948-4682. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.2202/1948-4682.1069.