# Automatic Sensitivity Identification with Generative AI

Ryan Gregson (2469038g)

April 19, 2024

## ABSTRACT

*Sensitive personal information is prevalent amongst documents within large collections that cannot be opened and made available without manual review to ensure the privacy and security of individuals. It would be beneficial to access the valuable information contained in these collections without infringing on privacy or confidentiality; however, the volume of content to review is too vast. This motivates the need for tools that can automatically identify and remove documents containing sensitive information. We propose a novel use of zero-shot learning with pre-trained LLMs, that generate natural language, to identify sensitive personal information and explore prompt engineering strategies to enhance the performance of these models. We find pre-trained LLMs are not successful at classifying sensitive personal information given basic classification instructions; however, using our prompt engineering strategies, LLMs make statistically significant improvements in identifying sensitive personal information compared to our baseline prompting approach. With our prompting strategies we achieve a balanced accuracy score of 67.96% on our test collection, with the LLM Mistral, a 14.58% improvement from basic prompts.*

## 1. INTRODUCTION

Large collections of digital documents contain sensitivities that must be managed and protected, such as sensitive personal information which can compromise an individual's safety. Archivists face the task of making these collections accessible safely [13]. Before these collections are made available, documents must be manually pruned by sensitivity review experts to ensure sensitive information is not released.

The increasing volume of digital documents makes conducting a fully manual sensitivity review less feasible. To alleviate reviewers, sensitivity classifiers [33, 31, 5] and review systems [37] to assist the sensitivity review process have been implemented and analysed. However, experts in the sensitivity review process still encounter challenges in navigating the extensive volume of digital content. Moreover, under the Freedom of Information Act 2000 (FOIA) [43], individuals are entitled to submit a freedom of information (FOI) request to government departments, local authorities, and other publicly funded bodies, yet national statistics show a growing number of delays in responding to these requests within the expected time limit [14, 9]. These delays damage relationships between public bodies and requesters, as limited access to information decreases the transparency of public bodies. The issues of navigating through potentially sensitive documents is likely to be exacerbated as digital archive collections expand, further challenging the feasibility of manual sensitivity review.

Therefore, technological advancements are necessary for protecting sensitive content and ensuring the sensitivity review process is more feasible than exhaustive manual (and technology-assisted) review. Despite this ideal reviewing tool, the importance of protecting sensitive information means all government documents reviewed in the foreseeable future will have some level of manual review [3]. This is reasonable as there is uncertainty around classifiers' capabilities. Identifying more reliable classifiers is an ongoing task, and studies show more accurate and confident classifiers benefit reviewers; increasing their reviewing speed when used within a review system [32].

Email collections can be enormous as emails are among the fastest-created types of digital documents due to regular and daily communication sent via this channel. Emails can be valuable to archives because they offer detailed insights into the development of ideas and help us understand communication, which enriches other research. Currently, many archives do not collect emails due to privacy concerns and the volume of documents in an email collection [40]. Companies may also collect emails to monitor breaches of company secrets [18]. Therefore, tools must be produced to identify sensitive personal information and remove documents to ensure individuals' privacy and security are maintained when exploring email collections. Aligning with current developments, we explore a novel approach using generative AI to perform sensitivity reviews to identify sensitive personal information, aiming to protect sensitive documents.

With the increased use of generative AI dominating natural language understanding tasks in other domains [49, 1, 51, 41], we aim to investigate the effectiveness of large language models (LLMs) in understanding and generating classifications and explanations concerning sensitive personal information within potentially sensitive email documents. In 2023, Baron et al. proposed the first use of a LLM to identify and explain sensitive information under FOIA Exemption 5 (the deliberative process privilege) [4]. Their approach uses the pre-trained LLM, ChatGPT-3.5 [6], and evaluates how prompt variations influence the model's responses for zero-shot learning. Our work extends the exploration of LLMs in sensitivity detection, specifically focusing on identifying sensitive personal information (exempt under FOIA Section 40) with generative LLMs, where to the best of our knowledge no research effort has been made. Our research aims to contribute to the ongoing advancement of these models, enhancing their capabilities in detecting sensitive personal information and facilitating automated sensitivity review.

There are multiple strategies to enhance the performance of LLMs. One method is fine-tuning, used to also enhance other neural models; however, this is very expensive for LLMs [38]. Furthermore, training datasets labelled with sensitivity annotations can be small or non-existent for some sensitive domains, resulting in weaker model tuning. Another recent strategy to improve the performance of LLMs is prompt engineering [27]. Prompt engineering aims to give pre-trained LLMs awareness of contextual nuances, emphasising focus on specific characteristics of the data. Successful research into zero-shot prompt engineering [56, 25] shows the potential for exploring this less resource-intensive prompting strategy when using LLMs to identify sensitive personal information.

This paper aims to explore if generative LLMs can improve on current methods for identifying sensitive personal information, and the effect of prompt engineering for sensitivity classification with LLMs. By leveraging LLMs and prompt engineering techniques we aim to show the possibility of automatic classification of sensitive information. With feasible (accurate and efficient) approaches, the scalability of handling and protecting sensitive personal information can be improved. In this paper we present a novel use of pre-trained LLMs with prompt engineering techniques to identify sensitive personal information.

We investigate prompts that add contextual information about the dataset and sensitive personal information, few-shot learning and chain-of-thought prompt engineering techniques with well-regarded open-source decoder-only LLMs Mistral [20], Mixtral [21], and Llama 2 [53]. Our experiments show that with the inclusion of our prompt engineering techniques, these generative LLMs become significantly more effective at classifying sensitive personal information; where Mistral performs best with prompt $S\_EC + PS + FS$ (context of possible sensitive email categories, description about personal sensitivity, and few-shot). Mistral with this prompt makes statistically significant improvements compared to our *Base* prompting strategy in sensitivity classification (McNemar's test $p < 0.05$), achieving a 14.58% balanced accuracy and 26.07% $\text{F}_2$-score increase.

The remainder of this paper is structured as follows. In Section 2 we present work relating to sensitivity classification and LLMs with prompt engineering. In Section 3, we motivate our dataset, model choice and prompting strategies, and in Section 4, we discuss our experimental setup. In Section 5 we present our results, then provide some further analysis in Section 6. Finally, we make our conclusions in Section 7.

# 2. BACKGROUND

In this section, we discuss related work of sensitivity detection, which uses sensitivity classifiers to identify sensitivities in potentially sensitive digital documents. These classifiers have traditionally used machine learning techniques such as support vector machines (SVMs) and involved feature engineering. Technology-assisted review (TAR) systems have utilised these classifiers to enhance the manual sensitivity review process. We then discuss large language models (LLMs): a deep learning technique used for natural language tasks. LLMs have advanced significantly, with research shifting to explore these models, where studies show performance near human-level reasoning and strong language understanding [41]. This motivates our project aims as LLMs have not been significantly explored to detect sensitive content and this task requires nuanced understanding of documents. Finally, we review prompt engineering and how this technique is used to enhance LLMs understanding. We believe there are prompting techniques and patterns from the literature that are relevant to the task of automatic sensitivity classification.

## 2.1 Sensitivity Detection

In 2014, the first classifier for digital sensitivity review was introduced to classify government records as sensitive regarding FOIA exemptions, namely Section 27 (International Relations) and Section 40 (Personal Information) [33]. Results from this study shows a text classification baseline could be improved with manually created features tailored for sensitivity detection. However, results also demonstrate that features such as country risk, relevant for international relations, negatively impacted classification of personal information sensitivities. This highlights the feasibility of sensitivity classifiers and reveals the requirement that features for different types of sensitivities must be considered separately. In our approach we use neural models; therefore, we do not have to manually create multiple feature sets for our sensitivities. Furthermore, recent trends have shown that deep learning models that work directly on the text tend to be more effective than traditional feature engineering approaches for most tasks [42]. This motivates our use of deep learning techniques for sensitivity classification.

Understanding the context of documents under review is important for correct sensitivity classification. McDonald et al. advanced their work of the first classifier for digital sensitivity review [33]: proposing extensions to text classification with part-of-speech features [30], as well as semantic (word embedding) features and additional term n-grams [31] to identify international relations and personal information sensitivities. Using a classifier with semantic features made significantly more accurate predictions by identifying latent sensitive relations in documents [31]. However, word embeddings only provide limited context about the overall sentence, whereas newer transformers facilitate contextual embeddings given the surrounding text [39]. One study using pre-trained extensions of BERT [7] (RoBERTa [28] and DeBERTa [16]) concludes these transformer methods perform better than traditional machine learning methods when classifying sensitive personal data [11]. These papers highlight a need for classifiers to be aware of the context of the document to capture subtle and latent sensitivities within sensitive documents. Therefore, we leverage these context-aware transformer-based language models in our sensitivity classification task. Additionally, context can be provided through prompting LLMs; therefore, we investigate how prompt engineering can provide necessary supplementary context for enhancing the identification of sensitivities.

Sensitivity review was once a fully manual approach; however, this has become infeasible as the number of digitally produced documents is ever increasing. Currently, sensitivity review systems with built-in sensitivity classifiers are used to assist expert reviewers, aiming to increase the throughput of reviewed documents. Analysis of these systems demonstrate reviewing accuracy and speeds increased given improved sensitivity classification accuracy [32]. Therefore, improving sensitivity classification is important for both assisting manual review, and pushing the boundary for a classifier that does not require final human involvement. In addition to potentially improving classification, these generative models have been treated as conversational agents

[46] and could also be explored as a personal assistant to expedite the review process. We aim to push the boundary; investigating if generative LLMs can classify sensitive documents. Research indicates that these models exhibit enhanced contextual understanding and reasoning capabilities [1]. Our objective is to explore whether LLMs can be effectively applied to identify sensitive content in documents and remove human involvement.

Recent research by Baron et al. has explored using a pre-trained LLM without any fine-tuning to identify government records that are exempt under FOIA Exemption 5 [4]. Baron et al. evaluated the zero-shot performance of the prominent language model ChatGPT-3.5 [6] across various prompts. Results from this experiment show that ChatGPT (as a classifier) is worse than the supervised text classification methods explored in their previous work using the same documents [5]. However, ChatGPT's generated text was deemed useful for reasoning about its classification choices. Baron et al. conclude that their confidence in sensitivity classification using ChatGPT is inconclusive, and they state this is the beginning of an investigation into generative AI to protect sensitive content [4]. This work lays a foundation for classifying sensitive content with generative LLMs and motivates exploration of prompt engineering. This importantly motivates our plan to use pre-trained generative LLMs as it demonstrates the decision-making abilities of generative LLMs in protecting sensitive content. Baron et al. use OpenAI's API for ChatGPT which is expensive. Therefore, our investigation will instead explore open-source LLMs (discussed further in Section 3.2) because they have no cost barrier. This means the models investigated in our research will be useful for individuals requiring extensive use of sensitivity classifiers; for example, archivists who aim to collect new digital documents such as emails without a large budget. We are also interested by the enhancements to model performance due to prompt engineering. However, Baron et al. do not consider notable prompt engineering techniques found in the literature to enhance model performance [57]. We explore the effect of more advanced prompting techniques for identifying sensitive content.

## 2.2 Large Language Models (LLMs)

Large language models (LLMs) have achieved excellent results in natural language processing tasks. These tasks can be split into categories such as general language understanding, question answering, sentiment analysis, named entity recognition, machine translation, and summarisation [49]. Expensive strategies, such as standard fine-tuning have been used to enhance model accuracy. However, recent consideration of prompt engineering shows this promising new paradigm can enhance model performance given no (or little) prior knowledge of the task beforehand; allowing effective zero-shot (or few-shot) learning with the LLM [27].

LLMs have been used successfully in document classification such as sentiment analysis tasks [12, 48, 2] and text classification tasks [54, 73]. A study comparing ChatGPT to state-of-the-art (SOTA) solutions shows ChatGPT outperforms the SOTA solution in a sentiment analysis task [3]. However, another thorough analysis of ChatGPT by Kocon et al. shows ChatGPT struggles most with emotional tasks [30]. Emotional tasks require a strong understanding of textual sentiment. Kocon et al. also show that for emotional tasks, such as identifying unhealthy conversations, ChatGPT still perceives sentiment more accurately than some human annotators. Hence, emotion classification is a difficult task; requiring pragmatic understanding of language that even SOTA models perform poorly at. Pragmatic tasks require the model to use additional knowledge that is not explicitly represented by distributional semantics [24]. As sensitivity classification requires understanding of the content in the document, this is a pragmatic task. Therefore, our approach employs prompting techniques to supplement the LLM in comprehending the question enquiring if sensitivity is present in the document, and for understanding the document to be analysed.

### 2.2.1 Prompting LLMs

Prompts are important in improving LLM responses. LLMs are effective with prompts as they resemble tasks that were solved during the original training process; for example, generating the next blanked word in the sentence [27]. In 2021, claims of shifting paradigms from 'pretrain, then fine-tune' to 'pre-train, then prompt' were stated [27]. Kocon et al state the *"performance of modern language models, such as T5, GPT-3, and ChatGPT, heavily relies on the quality of task-specific prompts,"* [24]. Basic prompting methods would provide a LLM with a question and the document text. We specialise prompts to assist the LLM in automatic sensitivity classification. Therefore, to produce high quality prompts for our task of sensitivity classification, we investigate different prompt engineering techniques to discover the most appropriate prompts.

Tuning-free prompting is a prompting method that does not change the parameters of the pre-trained LM; relying on the prompt to solely effect the possible answer from the model. Large closed-source models like ChatGPT [6] are used this way, where the model is frozen and users can only prompt it. This approach is referred to as zero-shot learning and is less expensive than fine-tuning a LLM. However, with no fine-tuning, heavy prompt engineering is necessary as prompts are the only way to describe the task to the model [27]. This means this task description must be stated coherently. Therefore, we rigorously investigate which prompt statements assist LLMs when classifying sensitivity.

Another prompting method, fixed-LM prompt tuning, can be seen to be better than tuning-free prompting due to achieving higher accuracy on benchmark tasks [27]. Fixing the prompt and fine-tuning the LM is explored in the literature and results show this method is effective for text classification [59]. However, this prompting method has the fine-tuning expense, and recent prompt engineering patterns [57] cannot be used as effectively as prompts are usually not human-interpretable [27]. Therefore, we focus instead on tuning-free prompting to see if recent prompt patterns can enhance LLMs when automatically classifying sensitive documents. Furthermore, small datasets can be more harmful when fine-tuning models [60], and most sensitive documents are not public unlike movie reviews for example. Thus, exploring tuning-free prompting evaluates if prompt engineering is effective for eliciting LLMs pre-trained knowledge to classify sensitive documents.

## 2.3 Prompt Engineering

Various prompt engineering techniques have been explored in the literature, with a prompt pattern catalogue recently published (discussing seventeen prompt patterns) by White et al. [57]. This catalogue focuses on prompts for conversational LLMs, such as chatbot ChatGPT [6]; however, many strategies can be used and adapted to our less interactive classification task. We believe the notable techniques relevant to the sensitivity detection task include: the cognitive verifier pattern, template pattern, and the context manager pattern.

**Cognitive Verifier Pattern:** Literature shows LLMs can reason better if a question is divided into additional smaller questions that help to answer the initial question [61]. The Cognitive Verifier Pattern is useful with ChatGPT, with the recommended prompt: "When I ask you a question, generate three additional questions that would help you give a more accurate answer. When I have answered the three questions, combine the answers to produce the final answers to my original question." [57]. This is beneficial for our task as the LLM may ask specific questions about the document, clarifying weak understanding which can lead to a more accurate classification. However, this pattern requires user input for every document as the questions may be different for each document, making it less feasible for our investigation of a fully automated approach.

**Prompt composition:** A similar strategy to the cognitive verifier pattern is prompt composition [27]. This method answers additional primer questions composed by us before answering the main question of classifying sensitivity. Instead of questions, we use sub-tasks for the LLM to acknowledge or reason about, which contribute to the main classification question [15]. Intermediate reasoning is used by the popular **Chain-of-Thought (CoT)** method [55, 8]. This method decomposes a problem into steps the model should take. It evokes 'thought' by asking the model to consider smaller separate questions at each step, and has been shown to improve LLMs' capability in complex reasoning tasks [55]. Sensitivity classification requires strong reasoning capabilities; therefore, we use this pattern to investigate if reasoning about sensitive personal information can improve the LLM's effectiveness compared to a baseline prompt.

**Answer engineering** is a strategy to constrain the response of the model [27]. In the literature, generative LLMs have used this technique when classifying text; for example, in sentiment classification tasks [59, 24]. White et al. regard this technique as the **Template Pattern** [57]. This pattern is useful as it allows efficient processing of the generated text. A disadvantage of constraining model output is that other useful output the LLM may have generated is avoided. Baron et al. do not enforce a strict output pattern when using ChatGPT to detect sensitive content [4]. Their study discusses that ChatGPT's explanations were its most interesting capability. However, our aim focuses on evaluating an automatic sensitivity classifier; therefore, we use answer engineering to control the output, retrieving the classification label for downstream tasks. Furthermore, although the output is constrained, this restriction only regards the first word generated. This means the model may continue to provide explanations that are useful, such as justifications for the classification which we can use in our manual analysis.

The **Context Manager Pattern** emphasises specific aspects of the task [57]. By emphasising information, this pattern assists the LLM by providing additional context about the document and the aim of the task. We believe this can better inform the LLM about what sensitive information should be identified, and using terms understood from pre-training can assist zero-shot classification. We know context is important for correct sensitivity classification, as from the literature sensitivity classifiers that utilised semantic features could identify latent sensitivities within sensitive documents [31], and transformer-based methods proved to be effective identifying sensitive personal data [11] (discussed in Section 2.1). Therefore, we aim to assess if the context manager prompt pattern can improve the effectiveness of LLMs for sensitivity classification by introducing context that directs the model to identify sensitivities we are interested in.

**Few-shot learning:** In the literature, a well-studied LLM training strategy is few-shot learning. When prompting, this is also known as in-context learning as the model parameters are not actually updated; instead, the model is given additional in-context information to process [12, 6]. This prompting strategy uses annotated samples within the prompt before the document of interest. Is has been shown this improves the effectiveness of LLMs and aligns generated text to expected output, which complements answer engineering for classification tasks. A concern is that LLMs have a limited context window; however, we choose models with an appropriate context window length for presenting example documents. Therefore, we explore if an in-context learning strategy improves sensitivity classification.

# 3. APPROACH

We discuss our approach of using a public email collection to explore the automatic classification of sensitive personal information in emails. We also discuss our motivations for using the open-source LLMs we choose for our experiment, then formally state our prompting strategies.

## 3.1 Sensitive Information in Emails

Identifying sensitive personal information is an important task, and previous work (discussed in Section 2.1) has aimed to classify personal information as it is the most prominent exemption that prevents a document from release in the UK [3]. Interviews by Iqbal et al. reveal interviewees are often concerned about the intermixing of private conversations and work emails [18]. They showed it is easy to accidentally reveal sensitive personal information within work emails as private information is commonly shared amongst our closest colleagues. Therefore, we explore sensitivity classification of work-related email documents. Automatic sensitivity classifiers could efficiently and safely inspect emails, then remove any containing sensitive personal information. Accessing email collections would also be useful for archives interested in storing information from government, universities and businesses for future research. However, currently email collections are rarely regularly collected by archives due to privacy concerns and the volume of documents in an email collection [40]. Improved sensitivity identification methods will allow archivists to open email collections while respecting the privacy of email authors, motivating our exploration of identifying sensitive personal information within emails using new techniques.

We use a public email collection, Enron [23], which was released during a legal investigation of the company's collapse in 2001. This collection contains 619,446 company email messages. A subset of 1702 email threads was annotated by students at UC Berkeley for relevance to eight coarse genres present in work emails [17]. These genres include Company Business and Strategy, Purely Personal, Personal but in Professional Context, Logistic Arrangements (such as meeting scheduling and technical support), Employment Arrangements, Document Collaboration, Empty Message (sending attachment), and Empty message (forwarded messages). Consequently, we believe our findings can be applied to other email collections in professional settings.

McKechnie et al. produce SARA [35], a collection of sensitivity-aware relevance assessments for UC Berkeley's Enron subset. McKechnie et al. use the coarse genres 'purely personal' which contain emails with no relation to work and discusses personal affairs, and 'personal but in a professional context' which contains emails that do have relation to work being done at Enron but discusses individuals' quality of work and personal opinions about employee treatment, to produce sensitivity labels for sensitive personal information. SARA's 1702 documents have 211 sensitive documents, 1491 non-sensitive documents, and is accessible through the python package 'IR datasets' [29].

## 3.2 Model Choice

Pre-trained LLMs are known to be effective zero-shot learners [25]. The benefit of zero-shot learning means we can have effective inference without the cost of fine-tuning. In our classification task, we use natural language to command the model to generate a class of interest that was not the primary focus during the pre-training stage. Our investigation aims to evaluate if LLMs can successfully generate the sensitive or non-sensitive class for a given document. Therefore, we investigate if LLMs are effective without any training as this suggests these pre-trained generative models will be useful in other sensitive domains. A limitation when using pretrained LLMs is that their training data will have included biases that are unwanted for assessing potentially sensitive documents. However, the large training process that comes with language models of this size means it is difficult to ensure all training is completed with unbiased data.

Recent generative LLMs, such as OpenAI's GPT-4 model [41], show great advancements on NLP benchmarks, showcasing the advanced reasoning capabilities of LLMs. However, accessing OpenAI's newest and high-performance models have a cost barrier. We instead utilise readily accessible models, which are open-source, and can be conveniently downloaded from Hugging Face [58] in our experiments. By using open-source LLMs we aim for easy extensibility by other researchers and affordability for everyone implementing sensitivity classification. Furthermore, many modern generative LLMs, such as GPT-4, use a decoder-only transformer architecture. Therefore, we have explored open-source decoder-only LLMs, including Meta's Llama 2 [53], and the Mistral models (Mistral [20] and Mixtral [21]). These models are often used in benchmarks, where the latest Mistral models show improved performance compared to Meta's Llama 2 models. Furthermore, Mixtral uses a mixture-of-experts paradigm [19, 52], which is becoming increasingly popular within generative LLMs and used by GPT-4.

We use versions of these LLMs that have been further instruction fine-tuned as these are better at generating responses to instructions and questions compared to their corresponding non-instruction counterparts [54]. Prompt engineering techniques are additional focused instructions; therefore, these models can respond more effectively given contextual information and queries about our sensitivity classification task.

Preliminary analysis exploring the knowledge these pretrained LLMs possess regarding sensitive personal information is demonstrated through a concise question answering experiment. To assess our models' understanding, we posed queries such as 'What does sensitive personal information mean?', and 'Describe UK FOIA Section 40'. The models effectively generated responses explaining the laws and regulations designed to safeguard sensitive personal information, such as the GDPR in Europe. Models also generated examples of protected attributes, including financial and health information, biometric and location data, as well as personal characteristics like race and religion (Appendix Table 6 & Table 7). The models' ability to produce contextually appropriate information based on their training data indicates a foundational understanding of sensitive personal information. This understanding gives us confidence in their application for our purposes of sensitivity review.

## 3.3 Prompting Strategies for Sensitive Personal Information

Prompt engineering is used to steer the model into being suitable for a task that is not necessarily the goal of initial fine-tuning objectives. Pretrained LLMs possess an extensive (English) vocabulary, and research indicates that these models, when appropriately prompted, can effectively identify and categorise documents in tasks like sentiment analysis [24, 26]. Moreover, prompts can influence the role or character the model assumes.

We design our prompts to instruct the model to use its understanding of sensitive personal information and directly apply this knowledge to identifying sensitive information within the email message. This allows the model to classify a document as sensitive or non-sensitive, similar to how a human reviewer would conduct a sensitivity review to protect sensitive content. Additionally, we can apply extra details about the task and email collection in the prompt to provide context to the model.

Emails in our chosen Enron collection are delivered from company email addresses where there is an expectation of professionalism. Additionally, SARA categorises sensitive emails as purely personal and personal but in a professional context. Describing these categories puts the sensitive information to identify in a suitable context, which helps explain our task to the LLMs. As well as simply providing context, we can also use few-shot prompting which provides examples of email messages and their annotation, showcasing sensitive personal information that has already been identified. We also apply more complex reasoning via chain-of-thought prompts, which allows the model to describe personal information within the message, and reason if it is sensitive, before classifying the message as sensitive or non-sensitive. This intermediary step serves as self-generated rationale, facilitating reasoning before the generated classification prediction.

# 4. EXPERIMENTAL SETUP

The objective of our experiments are to see how effectively we can classify potentially sensitive documents using generative LLMs, and the impact of prompt engineering strategies for identifying sensitive personal information. To do this we treat the combination of prompt engineering strategies

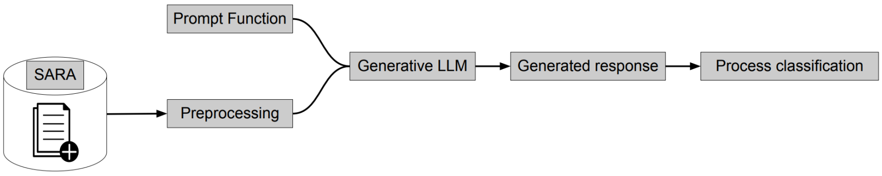


**Figure 1: Our experimental pipeline uses SARA from IR datasets, and combines processed documents with different prompt functions. This is given to a generative LLM in a zero-shot fashion, where we obtain generated classifications through natural language that are fit for processing.**

as prompt functions and use natural language to determine our classification predictions as shown by our experimental pipeline in Figure 1. In our experiment, we seek answers to the following questions:

- **RQ1**: How effective are zero-shot generative LLMs for identifying sensitive personal information within emails when given basic instructions?
- **RQ2**: Does prompt engineering improve the effectiveness of zero-shot generative LLMs for identifying sensitive personal information compared to a prompt with basic instructions?
  - RQ2.1: Does the context manager prompt engineering strategy, which introduces context about sensitive personal information, improve the effectiveness of sensitivity classification with generative LLMs?
  - RQ2.2: What is the effect of few-shot prompting on the effectiveness of sensitivity classification with generative LLMs?
  - RQ2.3: Does chain-of-thought use reasoning effectively to improve the effectiveness of sensitivity classification with generative LLMs?
- **RQ3**: Does our proposed use of generative LLMs for identifying sensitive personal information within emails improve on existing sensitivity classification methods for sensitive personal information?

## 4.1 Dataset

As the dataset for our experiments, we use the SARA collection [35]. The SARA collection has many long and unstructured email threads; therefore, we preprocess these documents to create consistency between the documents and remove noise across the collection, giving us more robust input to our LLMs. Our preprocessing involves:

1. Removing email headers, as we are interested in personal information found in the body of our messages. Furthermore, email headers results in more tokens, using LLMs' important space in the context window.
2. Removing email addresses from email thread metadata; replacing @ with a space. This obvious personal identifier inherent to email threads caused model confusion in early experiments. We remove these for identifying sensitive personal information within emails.
3. Removing all consecutive whitespace characters.
4. Using simple preprocessing strategies to remove numerical values, punctuation and lowercase our documents [50].

We do not remove any stop words from the documents as this is less representative of the training data seen by pretrained LLMs.

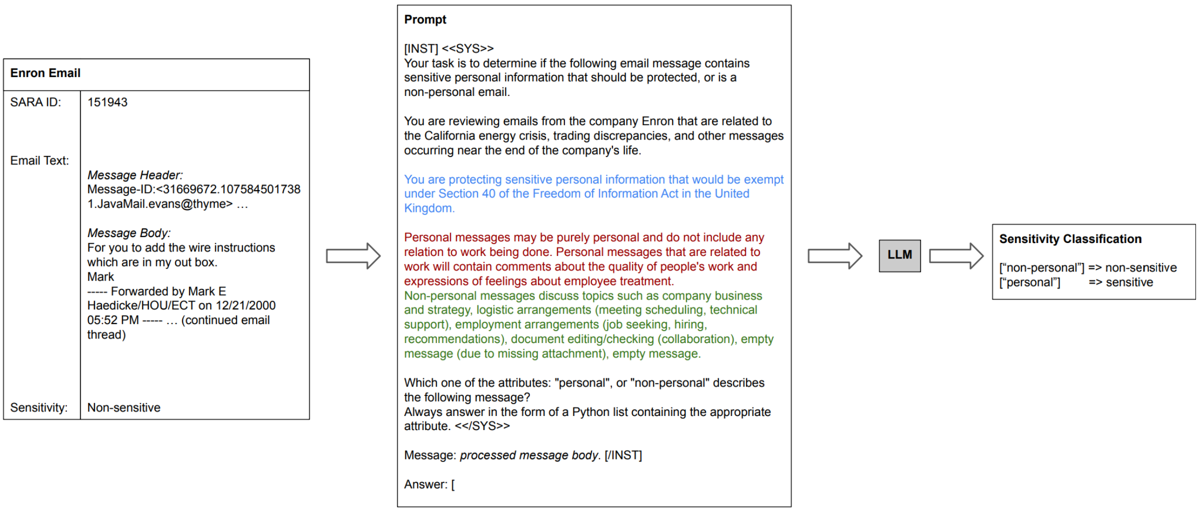
Preprocessing, such as header removal, leads us to remove 128 duplicate documents which have the exact same content but are sent to a different inbox directory for example. This leaves us with 1574 documents (196 sensitive, 1378 non-sensitive documents) to assess. As a sanity check, we extrapolate results for the removed documents using the kept identical document and find no significant changes to our results and concluded trends. We continue to process our documents by segmenting large documents because LLMs have a context window that limits the number of input tokens; therefore, we perform chunking on large documents at a token-level. Our model families, Llama-2 and Mistral, have a context window of 4096, and we segment our documents at 2048 tokens so we have available tokens for our prompt engineering methods. We find that many documents before chunking is applied are below this threshold, and we only require chunking for 191 documents. Overall, we have 1871 total documents used for inference with our LLMs. To acquire sensitivity predictions for our 1574 unique SARA documents, we collect documents that are segments of a larger document, where if any segment is identified as sensitive, the entire document is regarded as sensitive.

## 4.2 Tools

We use the deep learning framework, PyTorch [44], for development in our project. This Pythonic approach allows us to use Python libraries for preprocessing through to evaluation. We use the Hugging Face platform [58], which hosts open-source models for machine learning, to retrieve the LLMs for our experiment.

## 4.3 Prompt Design

Motivated by previous work (Section 2.3), we have designed different prompts following prompt engineering techniques. We refined our prompt through extending a text classification prompt used with generative LLMs [24, 47]. We identified what instruction and context must be provided to classify sensitive personal information within our documents, to establish a sensible base prompt. We establish our base prompt, consisting of an explanation of the problem (identifying sensitive personal information), information that these documents are email messages from Enron, a question we pose instructing the model to answer about the email containing sensitive personal information, and a prefix so the model response is appropriate for processing the classification label. This base prompt is shown by the black text in Figure 2. We then build on our base prompt using prompt engineering techniques, shown by coloured text



*Message Header:*
Message-ID:<31669672.107584501738 1.JavaMail.evans@thyme> …

*Message Body:*
For you to add the wire instructions which are in my out box.
Mark
----- Forwarded by Mark E Haedicke/HOU/ECT on 12/21/2000 05:52 PM ----- … (continued email thread)

[INST] <<SYS>>
Your task is to determine if the following email message contains sensitive personal information that should be protected, or is a non-personal email.

You are reviewing emails from the company Enron that are related to the California energy crisis, trading discrepancies, and other messages occurring near the end of the company's life.

You are protecting sensitive personal information that would be exempt under Section 40 of the Freedom of Information Act in the United Kingdom.

Personal messages may be purely personal and do not include any relation to work being done. Personal messages that are related to work will contain comments about the quality of people's work and expressions of feelings about employee treatment.
Non-personal messages discuss topics such as company business and strategy, logistic arrangements (meeting scheduling, technical support), employment arrangements (job seeking, hiring, recommendations), document editing/checking (collaboration), empty message (due to missing attachment), empty message.

Which one of the attributes: "personal", or "non-personal" describes the following message?
Always answer in the form of a Python list containing the appropriate attribute. <</SYS>>

Message: *processed message body.* [/INST]

Answer: [

LLM
Sensitivity Classification
["non-personal"] => non-sensitive
["personal"] => sensitive

**Figure 2: An Enron email message and our prompting strategies defined by coloured text. Our black text base prompt is extended using personal sensitivity context (blue), sensitive email category context (red), and non-sensitive email category context (green).**

in Figure 2.

We pose our question as a classification task: directly asking the model to choose one of two classes (personal or non-personal), as this direct class choice is done in previous work [24, 47]. Preliminary experiments found using classes "sensitive" and "non-sensitive" could not be understood effectively, generating explanations about sensitive business affairs; therefore, we use the classes "personal" and "non-personal" which is given context from our instruction to determine if the message contains sensitive personal information. As this question is posed as a direct choice of classes, we used 'Answer: [' as a prefix which was a sufficient guide to generate expected responses given instructions to include the attribute in a Python list, which is also done in related work as a form of answer engineering [24].

We use the **context manager prompt engineering pattern** to introduce additional context about sensitive email categories, and what should be treated as sensitive personal information. We provide background context on what sensitive personal information is using statements about Section 40 of the FOIA (shown in Figure 2 by blue text); motivated by prompts identifying information exempt under FOIA Exemption 5 [4], and from preliminary analysis demonstrating models could explain information about this act. We also use context about the dataset, stating information about the categories of emails within Enron. This context uses explanations of personal information that is purely personal and personal but in a professional context (sensitive email categories), as well as information about non-personal emails (non-sensitive email categories). We hypothesise adding context about personal information will inform the model about what we wish to protect and reduce confusion about sensitive documents by the model. We also hypothesise that information about the non-sensitive email categories, using Hearst's Enron annotations, will suggest to the model that it can differentiate emails into multiple sub-categories; for example, identifying a message belongs to the 'empty with attachment' category offers more nuance when understanding if short emails with less surrounding context are personal or non-personal. Overall, this additional prompted context aims to expand on using the simple and generic personal or non-personal keywords as seen in our base prompt.

We use **few-shot prompting** by presenting examples of sensitive and non-sensitive emails before the message we aim to classify. We use the same two examples for every document. Our sensitive example is purely personal which is the most important type of sensitivity to correctly identify, and our non-personal example is about arranging a work-related call which is common non-sensitive email within the collection. We also ensure these documents are concise such that they fit within the context window yet still convey why they are regarded as sensitive or non-sensitive.

We use the **chain-of-thought (CoT) prompt engineering technique**, as related work demonstrates this technique can improve LLMs at complex reasoning tasks. Fei et al. demonstrate three-hop CoT [8], where 'hops' are responses to subtasks that are fed back into the LLM to support answering harder questions. We hypothesise that three steps will be useful in our task by first identifying personal information directly from the text, reasoning about this found personal information being sensitive, then finally generating a classification label using this generated self-reasoning. The answer instructions for our three hops are:

1. State what personal information is present in the message if any.

2. You have already identified if personal information could be present within the message, which is shown after the message.

   State if the personal information you identified is sensitive personal information that should be protected.

3. You have already identified if personal information is present within the message, and then if this personal information is sensitive. This reasoning is shown after the message.

   Which one of the attributes: "personal", or "non-personal" describes the following message? Always answer in the form of a Python list containing the appropriate attribute.

, where each response is embedded into the next prompt. We also find separate hops are imperative to obtain final sensitivity classifications with our smaller and zero-shot models. Requested reasoning to be followed with a classification in a single instruction results in undesirable responses that cannot be processed; for example, reasoning with no final answer, or hallucinations such as short reasoning followed by generating a new email. This demonstrates that the instruction-based LLMs we use can struggle to follow instructions that are less direct and require sequential steps within a single prompt.

### 4.3.1 Prompt Abbreviations

We summarise these prompt strategies in Table 1 below, where multiple strategies used together are defined using addition notation.

**Table 1: Prompt strategies and abbreviations.**

| Prompt Strategy | Abbreviation |
|---|---|
| Basic Classification Instruction | Base |
| Context - Sensitive Email Categories | S_EC |
| Context - Non-Sensitive Email Categories | NS_EC |
| Context - Personal Sensitivity | PS |
| Few-shot | FS |
| Chain-of-Thought Hops | CoT |

## 4.4 LLMs

LLMs require GPU resources to efficiently operate. This limitation to computational cost leads us to choose models that are smaller than state-of-the-art. However, our models are still large, with 7 billion parameters for Llama 2 and Mistral, and 46.7 billion parameters for the larger Mixtral model, where we apply quantisation [10] to 4-bit given our resources. Our prompting techniques aim to show improvements to classification quality, so we hope these techniques will scale to larger models.

We use the popular open-source generative LLMs Llama-2-7B-Chat and Mistral-7B-Instruct-v0.2 at full precision as from early work we find these models are insufficient at the classification task when quantised; often hallucinating. For the larger model requiring quantisation we use Mixtral-8x7B-Instruct-v0.1 with post-training quantization [10] applied at 4-bits (Mixtral-8x7B-Instruct-v0.1-GPTQ).

In our experiments, we use 2 NVIDIA GeForce RTX 3090 GPUs made available through the University to conduct our experiments. Furthermore, we made inference to each document in a batch of one due to memory constraints.

We do not explore different generation hyperparameters within our experiment. We use default generation hyperparameters, but modify sampling to greedy decoding which is deterministic and produces reproducible results. The generation configuration also fixes the number of generated tokens; where we find ten tokens is the minimum number of suitable tokens to generate our classes. Our limits on generated tokens reduces the time for inference, and hence overall cost of using the models. Furthermore, for prompts of interest we generate up to a maximum of 150 tokens for analysis of model justification supporting the classifications through verbose explanations, and for chain-of-thought hops we generate up to 60 tokens for each hop before the final classification step.

## 4.5 Baselines

We set up simple baselines with scikit-learn [45] to compare our LLMs' classification performance against. We use baselines stratified random sampling and a naive most frequent classifier which is not useful for identifying sensitive personal information as sensitive documents are often the minority class. We use TF-IDF features for the traditional machine learning techniques logistic regression and support vector machines which have been used in previous sensitivity classification with SARA [35].

When conducting our statistical tests we treat our *Base* prompt as a baseline prompting comparison. Our hypothesis states that prompt engineering techniques should improve on a baseline prompting classification strategy. Furthermore we treat prompts without few-shot or chain-of-thought augmentation as pseudo-baselines to their related few-shot or chain-of-thought prompts within our statistical tests.

## 4.6 Evaluation Measures

In collections where sensitive documents are present, these are usually the minority class, and this is true for the SARA dataset with 12.4% of the collection labelled as sensitive. As this dataset is imbalanced, and we aim to identify sensitive documents, traditional accuracy is not the best metric due to its bias toward the majority class. Therefore, we select our main metrics as balanced accuracy (BAC) and the $F_2$-score. BAC is the average of the proportion of correct predictions in each class, and this ensures that the model is identifying this smaller proportion of sensitive documents which is critical. Moreover, a BAC of 0.5 demonstrates random predictions for binary classification. The $F_2$-score is a variation of the F-score that considers recall twice as important as precision, and this emphasises value of the model in protecting sensitive documents because the cost of misclassifying a sensitive document as non-sensitive (false negative) is greater than the cost of misclassifying a non-sensitive document as sensitive (false positive). We also report standard classification measures accuracy, precision, true positive rate (TPR), true negative rate (TNR) and $F_1$.

We use McNemar's non-parametric test [36] for significance testing. Our data is paired, and our outcomes are binary; therefore, we use McNemar's to conclude if one prompt has a statistically significant better performance ($p < 0.05$) given a fixed model. McNemar's is calculated from the contingency tables of the prompt strategies we compare. Significant improvements compared to the baseline prompt (*Base*) are denoted by †. Additionally, significant improvements to pseudo-baselines compared to augmentation with few-shot or CoT prompting are denoted by ‡.

## 4.7 Limitations

A limitation of our exploration of prompt engineering is that prompts can be brittle and must be crafted carefully to ensure relevant instructions are given to the model. New work by Khattab et al. aims to optimise prompt instructions using DSPy teleprompters [22]. Teleprompters search and craft prompts given an initial base prompt instruction and a small sample of data. This strategy may help to discover information useful in prompts that we do not explicitly state in our engineered prompts. However, for complex tasks that require direction and pragmatic understanding, we believe our stated instructions would complement these optimisation approaches to potentially find the best formatted prompt instructions for use in sensitivity identification.

Another limitation is the pre-trained biases and understanding of information that the pretrained LLM is most attuned to. For example, LLM responses initially understood FOIA within the context of American law, unable to find Section 40, encouraging us to explicitly state United Kingdom within this prompt context. Furthermore, large private collections may have language that is not similar to information seen during model training. For example, SARA is a collection from the early 2000s, and email communication has evolved. Therefore, we suggest that LLMs that experience fair and diverse training will be less limited at identifying sensitive information given appropriate prompts.

Finally, a limitation of using smaller models refrains our exploration of the upper performance of zero-shotting generative LLMs. Research shows that a larger model size results in significant improvements compared to smaller models. Despite this, we believe it is important to explore these smaller models as token digestion and generation using large models have expensive API costs, or require expensive hardware to load these models into memory which may not be feasible.

## 5. RESULTS

Our results discuss the feasibility of using zero-shot generative LLMs to automatically identify sensitive personal information, and what strategies improve their performance. Table 2 contains the results of our different prompting strategies with models Mistral, Mixtral, and Llama 2 for SARA.

**RQ1. How effective are zero-shot generative LLMs for identifying sensitive personal information within emails when given basic instructions?**

We first analyse the effectiveness of different generative LLMs given a base classification prompting strategy that is motivated by the literature. The top row of Table 2 shows the performance of the *Base* strategy. Using our main metrics we see Mistral with 0.5338 BAC and 0.1908 $F_2$, 0.5244 BAC and 0.3423 $F_2$ for Mixtral, while Llama 2 obtains 0.5492 BAC and 0.4298 $F_2$. These scores are better than entirely naive random classifiers which do not protect many or any sensitive documents effectively; however, compared to other machine learning techniques in the literature this base prompt is ineffective at sensibly performing sensitivity review [35]. Furthermore, different models behave differently given this same base prompt instruction: Mistral is ineffective at correctly identifying sensitive documents, shown by a low TPR of 0.1939, whereas Llama 2 is overly protective of sensitive documents as shown by the highest TPR of 0.9184. Mixtral makes judgements between these models, misclassifying both classes in similar proportions. Overall, in answer to RQ1, we say that generative LLMs are ineffective in detecting sensitive personal information given a simple base prompting strategy.

**RQ2. Does prompt engineering improve the effectiveness of zero-shot generative LLMs for identifying sensitive personal information compared to a prompt with basic instructions?**

We now analyse the impact and effectiveness of each prompt engineering strategy, then answer how effective our prompt engineering techniques were overall across the entire experiment.

**RQ2.1. Does the context manager prompt engineering strategy, which introduces context about sensitive personal information, improve the effectiveness of sensitivity classification with generative LLMs?**

We introduced three forms of context: sensitive and non-sensitive email categories ($S\_EC$, $NS\_EC$), and a personal sensitivity definition ($PS$). Table 2 shows the performance of these prompt engineering techniques given different combinations. We see that our contextual phrases within prompts does improve the classification effectiveness, with significant differences across all strategies except $PS$ with Mistral and Llama 2. However, the addition of $PS$ with $S\_EC$ and $S\_EC + NS\_EC$ does further improve the classification performance of these prompts compared to the exclusion of $PS$ with Mistral and Mixtral; therefore, we suggest this context is useful. For Mistral and Mixtral, the combination of all three contextual prompts achieves the best classification performance, and Llama 2 performs best with just the $S\_EC$ context. Therefore, we conclude our additional context about sensitive personal information and email categories suitable for professional email collections lead to statistically significant improvements to zero-shot sensitivity classification.

We believe prompt instructions can be used as natural language features for what justifies sensitive personal information within an email, because this context has improved classification of sensitive documents. This effect is true for Mistral and is shown by the increasing TPR. Furthermore, we find using prompt instructions explaining other types of documents that would be present within the document collection further enhances sensitivity classification. We see this from the increase of TNR compared to approaches without email category context and improvements to the precision metrics (mitigating model confusion). Interestingly, the inclusion of non-sensitive email categories improves the classification of sensitive email documents, where we see improvements to TPR for Mistral and Mixtral comparing $S\_EC$ and $NS\_EC$ prompting approaches. We believed this contextual information would in fact mitigate false positives by exposing the model to be aware of the type of document it is classifying, which in turn may have introduced more true (and false) negatives. We conclude this contextual information helps to guide the model in classification alongside its pre-trained knowledge of sensitive personal information by providing both informative focus to sensitive personal information, and further knowledge about the potential contents of the email documents within the system context.

Overall, in answer to RQ2.1, the use of the context manager strategy improved the effectiveness of using generative LLMs in zero-shot sensitivity classification.

**RQ2.2. What is the effect of few-shot prompting on the effectiveness of sensitivity classification with generative LLMs?**

From Table 2, we see augmentation with few-shot ($FS$) improves the effectiveness of Mistral for identifying sensitive personal information. For every prompt except $S\_EC + NS\_EC$, Mistral with few-shot makes a statistically significant improvement to comparable zero-shot prompts, and all BAC and $F_2$ values are seen to increase given few-shot. For Mixtral and Llama 2, the addition of few-shot makes

**Table 2: Results for combinations of different prompting strategies, compared against the baseline prompting strategy. We embolden the overall highest value for each metric, and underline the highest value for each metric for each prompt strategy.**

| Prompt | | Model | Accuracy | Precision | TPR | TNR | $F_1$ | $F_2$ | BAC |
|---|---|---|---|---|---|---|---|---|---|
| Base | | Mistral | 0.7891 | 0.1792 | 0.1939 | 0.8737 | 0.1863 | 0.1908 | 0.5338 |
| | | Mixtral | 0.5006 | 0.1349 | 0.5561 | 0.4927 | 0.2171 | 0.3423 | 0.5244 |
| | | Llama 2 | 0.2719 | 0.1374 | **0.9184** | 0.1800 | 0.2390 | 0.4298 | 0.5492 |
| S_EC | † | Mistral | 0.8653 | 0.4216 | 0.2194 | 0.9572 | 0.2886 | 0.2427 | 0.5883 |
| | † | Mixtral | 0.5299 | 0.1699 | 0.7143 | 0.5036 | 0.2745 | 0.4353 | 0.6090 |
| | † | Llama 2 | 0.5299 | 0.1699 | 0.7143 | 0.5036 | 0.2745 | 0.4353 | 0.6090 |
| S_EC+NS_EC | † | Mistral | 0.8564 | 0.4038 | 0.3214 | 0.9325 | 0.3580 | 0.3351 | 0.6270 |
| | † | Mixtral | 0.5616 | 0.1873 | 0.7551 | 0.5341 | 0.3002 | 0.4701 | 0.6446 |
| | † | Llama 2 | 0.6296 | 0.1670 | 0.4949 | 0.6488 | 0.2497 | 0.3553 | 0.5718 |
| PS | | Mistral | 0.7827 | 0.2148 | 0.2806 | 0.8541 | 0.2434 | 0.2644 | 0.5674 |
| | † | Mixtral | 0.5394 | 0.1411 | 0.5306 | 0.5406 | 0.2229 | 0.3419 | 0.5356 |
| | | Llama 2 | 0.2605 | 0.1350 | 0.9133 | 0.1676 | 0.2352 | 0.4242 | 0.5404 |
| S_EC+PS | † | Mistral | 0.8653 | 0.4467 | 0.3418 | 0.9398 | 0.3873 | 0.3587 | 0.6408 |
| | † | Mixtral | 0.6436 | 0.1973 | 0.6071 | 0.6488 | 0.2979 | 0.4290 | 0.6280 |
| | † | Llama 2 | 0.5273 | 0.1667 | 0.6990 | 0.5029 | 0.2692 | 0.4265 | 0.6009 |
| S_EC+NS_EC+PS | † | Mistral | 0.8571 | 0.4249 | 0.4184 | 0.9194 | **0.4216** | 0.4197 | 0.6689 |
| | † | Mixtral | 0.6404 | 0.2100 | 0.6837 | 0.6343 | 0.3213 | **0.4712** | 0.6590 |
| | † | Llama 2 | 0.5521 | 0.1460 | 0.5357 | 0.5544 | 0.2295 | 0.3493 | 0.5451 |
| Base+FS | † ‡ | Mistral | 0.8520 | 0.3609 | 0.2449 | 0.9383 | 0.2918 | 0.2617 | 0.5916 |
| | † ‡ | Mixtral | 0.8259 | 0.2926 | 0.2806 | 0.9035 | 0.2865 | 0.2829 | 0.5920 |
| | † ‡ | Llama 2 | 0.8564 | 0.2222 | 0.0612 | 0.9695 | 0.0960 | 0.0716 | 0.5154 |
| S_EC+FS | † ‡ | Mistral | 0.8405 | 0.3799 | 0.4439 | 0.8970 | 0.4094 | 0.4294 | 0.6704 |
| | † ‡ | Mixtral | 0.8640 | 0.4392 | 0.3316 | 0.9398 | 0.3779 | 0.3487 | 0.6357 |
| | † ‡ | Llama 2 | 0.8634 | 0.3908 | 0.1735 | 0.9615 | 0.2403 | 0.1952 | 0.5675 |
| S_EC+NS_EC+FS | † | Mistral | 0.8437 | 0.3821 | 0.4133 | 0.9049 | 0.3971 | 0.4066 | 0.6591 |
| | † ‡ | Mixtral | 0.8736 | 0.4835 | 0.2245 | 0.9659 | 0.3066 | 0.2514 | 0.5952 |
| | † ‡ | Llama 2 | 0.8717 | 0.4605 | 0.1786 | 0.9702 | 0.2574 | 0.2035 | 0.5744 |
| PS+FS | † ‡ | Mistral | 0.8374 | 0.3469 | 0.3469 | 0.9071 | 0.3469 | 0.3469 | 0.6270 |
| | † ‡ | Mixtral | 0.8602 | 0.2931 | 0.0867 | 0.9702 | 0.1339 | 0.1010 | 0.5285 |
| | † ‡ | Llama 2 | 0.8488 | 0.1613 | 0.0510 | 0.9623 | 0.0775 | 0.0591 | 0.5066 |
| S_EC+PS+FS | † ‡ | Mistral | 0.8259 | 0.3545 | 0.4847 | 0.8745 | 0.4095 | 0.4515 | **0.6796** |
| | † ‡ | Mixtral | 0.8710 | 0.4667 | 0.2500 | 0.9594 | 0.3256 | 0.2756 | 0.6047 |
| | † ‡ | Llama 2 | 0.8532 | 0.3267 | 0.1684 | 0.9507 | 0.2222 | 0.1864 | 0.5595 |
| S_EC+NS_EC+PS+FS | † ‡ | Mistral | 0.8342 | 0.3663 | 0.4541 | 0.8882 | 0.4055 | 0.4333 | 0.6712 |
| | † ‡ | Mixtral | **0.8761** | **0.5079** | 0.1633 | **0.9775** | 0.2471 | 0.1889 | 0.5704 |
| | † ‡ | Llama 2 | 0.8494 | 0.3566 | 0.2602 | 0.9332 | 0.3009 | 0.2751 | 0.5967 |
| S_EC+NS_EC+PS+CoT | † ‡ | Mistral | 0.5667 | 0.1373 | 0.4694 | 0.5806 | 0.2125 | 0.3164 | 0.5250 |
| | † ‡ | Mixtral | 0.7395 | 0.1677 | 0.2755 | 0.8055 | 0.2085 | 0.2441 | 0.5405 |
| | † ‡ | Llama 2 | 0.3259 | 0.1236 | 0.7245 | 0.2692 | 0.2112 | 0.3673 | 0.4969 |

statistically significant differences from the pseudo-baseline zero-shot prompt; however, few-shot does not improve every prompt instance of these models. Mixtral and Llama 2 suffered from low specificity with our zero-shot prompts; however, with few-shot prompting the opposite is true, where TNR is extremely high, greater than 0.9000 for all few-shot prompts. With a high TNR, one of our main metrics $F_2$ is extremely low for these model-prompt settings; therefore, we conclude few-shot with Mixtral and Llama 2 is not effective for identifying sensitive personal information.

We see that few-shot prompt engineering consistently improves sensitivity classification with Mistral only. Therefore, we conclude this model is more effective than Llama 2 and Mixtral with few-shot prompting. Furthermore, we also speculate that the poor performance with the other models using few-shot could be for reasons such as poor example choices and example ordering.

We included two examples in-context: one sensitive, and one non-sensitive. As the second example was always non-sensitive and our causal LLMs only use previous context, prioritising more recent tokens, the fact that an answer with a non-sensitive label appears more recently may be confusing the model. As messages are reasonably long, the examples are long, which creates a greater distance between demonstrated answers. This could impact the usefulness of few-shot prompting. Furthermore, we conducted a short experiment during post-hoc analysis, to assess if class ordering impacts few-shot prompting. We found insignificant differences using a single model-prompt setting: Mixtral with $S\_EC + NS\_EC + PS$. We chose this model and prompt because of the large decrease compared to the zero-shot prompt without few-shot examples in our results. Despite this, due to the brevity of our post-hoc experiment, we cannot conclude that the reordering of examples has no impact.

As well as example ordering, the choice of our examples could influence our results. Emails in the collection that are non-sensitive are relevant to six categories as shown by Hearst [17] and Mckechnie [35]. Introducing multiple email categories using the context manager pattern improved sensitivity classification. Therefore, it could be useful to investigate using examples of multiple sensitive and non-sensitive email categories to improve the effectiveness of sensitivity classification compared to the examples chosen. We also consider that as emails could be structured the same way by individuals, retrieving 'best' examples will be challenging.

Furthermore, ongoing research is investigating novel few-shot prompting strategies. Techniques such as DSPy's bootstrapped few-shot which generate synthetic and representative few-shot samples could be explored. This technique could be used to release prompts with examples publicly as synthetic few-shot examples may be useful within other collections.

**RQ2.3 Does chain-of-thought use reasoning effectively to improve the effectiveness of sensitivity classification with generative LLMs?**

For Chain-of-Thought (CoT), our models do not behave as we hypothesised, performing worse than no CoT. We believed taking reasoning steps would elicit sensible self-explanations to better classify the email message as sensitive or non-sensitive given additional context of personal information within the email.

There is a statistically significant difference between zero-shot and CoT prompts for all of our models where the zero-shot prompt outperforms CoT techniques, shown by BAC lowering by 14.39%, 11.85% and 4.82% for Mistral, Mixtral and Llama 2 respectively. In fact, CoT for Mistral and Llama 2 performs worse than the *Base* prompt; Llama 2 even performing worse than a naive random classifier. We believe our self-reasoning step did not highlight useful personal information, often only identifying names which caused model confusion compared to the exclusion of this step. Overall, we conclude that our proposed CoT strategy is ineffective for improving sensitivity classification.

**RQ2 Overall.**

Considering the effectiveness of our prompt engineering strategies across the entire experiment, we observe that Mistral models outperform Llama 2. Mistral obtains its highest BAC and $F_2$ with prompt $S\_EC + PS + FS$ at 0.6796 BAC and 0.4515 $F_2$, and Mixtral with prompt $S\_EC + NS\_EC + PS$ achieves 0.6590 BAC and 0.4712 $F_2$, whereas Llama 2 (with prompt $S\_EC$) achieves 0.6090 BAC and 0.4353 $F_2$ at its best. Mistral's performance over Llama 2 is also in agreement with popular benchmark tasks [20]. We conclude Mistral models are better at our task of identifying sensitive personal information. Interestingly, the Mixtral model that is larger than Mistral does not outperform Mistral in sensitivity classification. We believe there are some differences between these models regarding the overall effectiveness of sensitivity identification. Each model with its best prompt setting has significant differences, where they disagree on 496 documents: Mistral correctly classifying 394 documents (9 sensitive, 385 non-sensitive) and Mixtral correctly classifying 103 different documents (48 sensitive, 54 non-sensitive) that each other could not correctly classify. This explains Mixtral's higher $F_2$-score as the more recall-oriented model is protecting more sensitive documents.

The Mistral and Llama 2 models are both implemented using the transformer decoder architecture. However, our results show that Mistral outperforms Llama 2 over multiple conditions, except our *Base* and $S\_EC$ prompt. Llama 2 is highly confused during classifying sensitive personal information in SARA, and this is shown by low specificity (TNR) scores given zero-shot prompts. Mixtral's shortfalls follow the same trend as Llama 2; however, incorrect classifications are less severe with Mixtral. Alternatively, Mistral is more rationale at detecting sensitive personal information, and this is shown by the greater $F_1$-score across all prompts except *Base*. We suggest that the Mistral LLM has better decision-making capabilities when identifying sensitive personal information, and that Llama 2 is over-protective of personally identifiable information as analysis of responses show that Mistral can deduce that personal information is not always sensitive personal information, which Llama 2 struggles with more. This is important because models that overpredict sensitivity, by treating names as sensitive personal information for example, are less effective as sensitivity classifiers. They fail to distinguish between personal information inherent to email documents, which should generally be considered non-sensitive, and truly sensitive personal information that is of interest. We do not have insights into the training data of these models due to the current competitive nature of pre-training LLMs; however, Meta has aimed to censor toxic generation from their models, hence may be over-protective to attributes such as names. Overall, despite some shortcomings of the generative LLMs, we conclude that applying prompt engineering did improve the overall effectiveness of sensitivity classification due to improved metrics compared to *Base* prompt approaches. Furthermore, we conclude that we are most confident in the Mistral LLM as this obtains the highest BAC score which is one of our main evaluation measures, as well as a good $F_2$ score compared to other approaches.

**RQ3. Does our proposed use of generative LLMs for identifying sensitive personal information within emails improve on existing sensitivity classification methods for sensitive personal information?**

We compare sensitivity classifications made by our generative LLMs to naive baselines and traditional machine learning techniques used in previous work [35]; following their downsampling training strategy that uses a train-test split of 20:80. These traditional approaches are shown in Table 3, which also includes Mistral with a *Base* prompting approach, and Mistral with the best performing prompt strategy, $S\_EC + PS + FS$, where we exclude the samples used as training data by our trained classifiers when calculating metrics. We observe that our baseline prompt performs slightly better than our naive baselines stratified random sampling (Random) and most frequent (MF) classification. However, with our best performing system prompt, we

**Table 3: Results for SARA's test set, using traditional approaches, Mistral-$Base$, and Mistral-$S\_EC + PS + FS$.**

| Strategy | Acc | Prec | TPR | TNR | $F_1$ | $F_2$ | BAC |
|---|---|---|---|---|---|---|---|
| Random | 0.4976 | 0.1291 | 0.5062 | 0.4964 | 0.2058 | 0.3196 | 0.5013 |
| MF | **0.8714** | 0.0000 | 0.0000 | **1.0000** | 0.0000 | 0.0000 | 0.5000 |
| SVM | 0.7651 | 0.2919 | 0.5802 | 0.7923 | 0.3884 | 0.4845 | 0.6863 |
| LR | 0.7468 | 0.2873 | **0.6543** | 0.7605 | 0.3992 | **0.5211** | **0.7074** |
| Mistral-Base | 0.7825 | 0.1782 | 0.1914 | 0.8698 | 0.1845 | 0.1886 | 0.5306 |
| Mistral-Best | 0.8230 | **0.3620** | 0.4938 | 0.8716 | **0.4178** | 0.4603 | 0.6827 |

achieve more accurate sensitivity classifications, where our generative LLMs alongside prompt engineering do perform significantly better than both baselines. Machine learning techniques logistic regression (LR) and support vector machine (SVM) perform similar to our best performing model settings where LR has BAC 0.7074 and $F_2$ 0.5211 (both highest), SVM has BAC 0.6863 and $F_2$ 0.4845, and zero-shot Mistral with $S\_EC + PS + FS$ has BAC 0.6827 and $F_2$ 0.4603 on our separated test collection. Supervised machine learning strategies have more effective recall than the Mistral models, making them more effective at identifying and protecting sensitive documents. Notably, from our main results table (Table 2), the top performing Mixtral prompt identifies more sensitive documents than the traditional machine learning approaches as shown by a higher TPR of 0.6837. Moreover, this best Mixtral prompt also has 0.6590 BAC across the whole collection, demonstrating sensible sensitivity classifications are still made; unlike model-prompt variations that classify nearly every document as sensitive and hence have a very high TPR but are not useful. However, this Mixtral model prompt setting still performs worse than our traditional machine learning strategies considering our two main metrics BAC and $F_2$ due to its limitation of over-predicting sensitivity.

Overall, in answer to RQ3, we say that it is noteworthy that these generative LLMs can effectively classify sensitive personal information without training; however, our approach does not significantly improve on current machine learning approaches such as logistic regression and SVMs, instead obtaining similar results to these classifiers when comparing against our best performing model-prompt setting Mistral-$S\_EC + PS + FS$. Therefore, their utility in sensitivity classification should be considered carefully due to their more expensive computational cost compared to machine learning techniques.

# 6. ANALYSIS

Our results show that our prompting strategies do improve sensitivity classification, where McNemar's statistical tests show these strategies produce statistically significantly different classifications from a base prompting strategy. For the best performing model, Mistral, these are significant improvements in sensitivity classification. Figure 3 displays the gains and losses of correct predictions for each individual class for our prompting strategies compared to baseline approaches. We see our gains given our prompting strategies is greater than our losses compared to a baseline with Mistral (Figure 3a). Furthermore, few-shot alongside our prompting strategies with Mistral demonstrate gains are correctly identifying sensitive documents, albeit at the cost of now misclassifying some non-sensitive documents as sensitive (Figure 3d). As our objective aims to identify and protect sensitive personal information, we can justify that Mistral's performance makes it a feasible sensitivity classifier. Analysis of the gains made by Mixtral with our prompting strategies shows encouraging improvements to sensitivity classification, where improvements are made to both non-sensitive and sensitive correct classification when zero-shot (Figure 3b). However, with few-shot applied to Mixtral, the losses in classifying sensitive documents are noticeably poor, where the model is much more likely to classify non-sensitive documents correctly instead (Figure 3e). Finally, (Figure 3c and Figure 3f) show Llama 2's significant issues with correctly identifying sensitive personal information, and our prompt engineering efforts improve non-sensitive classifications. However, the high TPR of Llama 2 $Base$ gives reason for the loss in correctly identified sensitive documents.

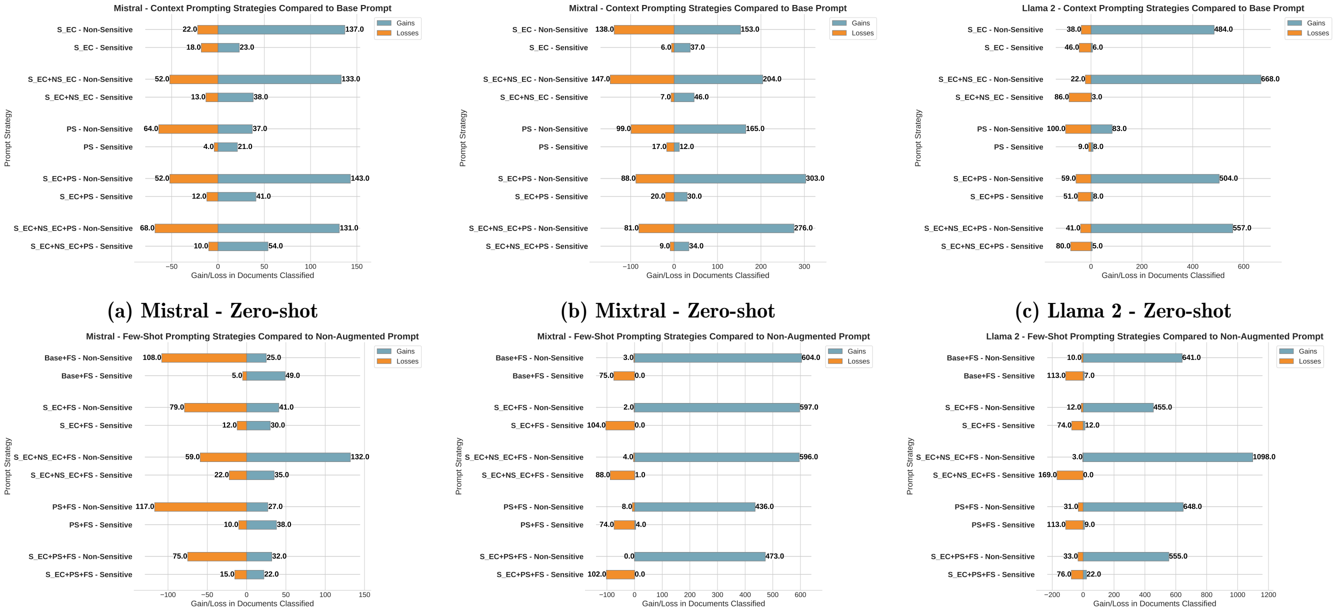
## 6.1 Misclassified documents

### 6.1.1 *Entirely Misclassified*

**Table 4: Entirely misclassified documents**

| Email Content | Model Response |
|---|---|
| http://www.pserc.wisc.edu/index_home.html | ["non-personal"]. This message appears to be a URL and does not contain any personal information. |
| http://www.ssrn.com/ | ["non-personal"]. The message does not contain any personal information or comments related to work or employees. It is a simple URL link. |
| http://www.housingfinance.org/Country%20Specific%20Information/country_specific_information.htm | ["non-personal"]. This message appears to be a URL and does not contain any personal information. |

From our results, Llama 2 with a $Base$ prompt achieves an extremely high TPR of 0.9184, whereas the inclusion of few-shot appears to switch most predictions, where Table 2 shows Llama 2 with $Base + FS$ at an extremely high TNR of 0.9695 and low TPR of 0.0960. These two prompts, base and base augmented with few-shot already covers many documents lowering the number of fully misclassified documents and demonstrating Llama 2's lack of confidence at classifying sensitive documents confidently when zero-shot. Moreover, we have 3 sensitive documents that are entirely misclassified by all model-prompt settings. Inspecting these 3 emails, we see they are each single message threads where the message body is only a shared link. Hearst's Enron email annotations categorise these emails as personal but in a professional context. Our verbose responses of our best classifier, Mistral with $S\_EC + NS\_EC + PS + FS$ (which also correctly identifies the most sensitive documents across all prompt strategies with Mistral), are shown in Table 4. Other prompt-model settings with verbose generation for these documents respond similarly; stating URLs are non-sensitive, or that they have not been given enough information to conclude there is sensitive personal information within the message, hence justify this as non-personal. We note some verbose responses were able to correctly identify SSRN is an acronym



(a) Mistral - Zero-shot

(b) Mixtral - Zero-shot

(c) Llama 2 - Zero-shot

(d) Mistral - Few-shot

(e) Mixtral - Few-shot

(f) Llama 2 - Few-shot

**Figure 3: Comparative results of zero-shot against $Base$ prompt, and few-shot prompting against non-augmented counterpart prompts. We include all three models and plot the gains and losses of each individual class.**

for 'Social Science Research Network' but did not show any other interesting understanding beyond this. Given these emails are very short, this demonstrates that LLMs can struggle with limited surrounding context and could benefit from more external context. We notice an interesting feature within the message header when inspecting these full documents: the sender and receiver are the same person using a different email address (from j.kaminski@enron.com, to vkaminski@aol.com), which are addresses owned by Vincent Julian Kaminski, a managing director at Enron. There are two other very similar emails, also sent by Vincent to himself and containing a single hyperlink: *'http://www.vaionl-ine.it'* and *'http://www.housingfinance.org/Country%20Spec-ific%-20Information/Fact%20book%20(e)%2099.pdf'*, which both have misclassification rates of 0.9487. Llama 2 with prompts $Base$ and $Base + PS$ make correct predictions for these hyperlinks; however, we have concluded that Llama 2 given these prompting strategies lacks the strength of a successful sensitivity classifier by not identifying the nuanced sensitive personal information as both experimental settings achieve an extremely high TPR and low precision meaning nearly every document is classified as sensitive. However, both strategies also do not include our guided definition of sensitive personal information expected in emails ($S\_EC$). This may allow them to identify sensitive information because they are not being instructed about the sensitive categories of interest within this work-related collection.

From our inspection of entirely misclassified documents, we suggest that email header metadata could be useful for classifying sensitive personal information; in particular, the email correspondents. Further context could be given to the model, such as the number of correspondents and their relationships if this is known or mined in advance. However, from our CoT experiment we observed that identifying personal information, such as names, can cause confusion. Therefore, it is necessary to define relationships in a way that does not overwhelm the context; for example, strong relationships may influence the model. We also conclude that sensitive content can be embedded within email documents that cannot be identified from a purely textual perspective, such as URLs. We note there are also documents with JPGs that LLMs often misclassified, not realising they were sensitive. Therefore, we suggest awareness of non-textual elements could be important in correctly identifying sensitive information.

### 6.1.2 *Top 50 Misclassified Documents*

We evaluate the top 50 overall misclassified documents (where other models top 50 misclassified messages share the overall top 50 with Mistral at 31, Mixtral at 30 and Llama 2 at 20). From Table 5 we see a larger proportion of sensitive documents are misclassified across all classifiers, and that there are 8 non-sensitive and 42 sensitive documents within the top 50 misclassified messages overall.

Non-sensitive documents that are misclassified discuss company business and strategy (which is the most common non-sensitive email category) and employment arrangements. Moreover, the employment arrangement emails make direct criticisms and opinions about individuals. Inspecting part of one of Mistral's verbose responses: *"it is important to note that the message contains negative comments about specific individuals, which may be considered sensitive or confidential if shared outside of a small circle of trusted colleagues"*, demonstrates that our prompting strategy to provide context about sensitive email categories using McKechnie's utilisation of personal information within a professional context [35] has confused the LLM. This sentiment towards a colleague has a non-sensitive ground truth, and our instructions to identify comments about colleagues and employee treatment have likely encouraged the LLM that these opinions are sensitive. The exclusion of our prompt explanation of sensitive categories, with prompt strategy $PS$, correctly identifies this email. However, we believe prompting with $S\_EC$ context provides important information to create more sensible

**Table 5: Top 50 misclassified documents**

| Model | All Wrong | | Top 50 | |
|---|---|---|---|---|
| | Non-Sens | Sens | Non-Sens | Sens |
| Mistral | 6 | 45 | 6 | 44 |
| Mixtral | 15 | 28 | 18 | 32 |
| Llama 2 | 4 | 4 | 26 | 24 |
| All | 0 | 3 | 8 | 42 |

sensitivity classifications as shown by the overall gains in prediction correctness (Figure 3).

There is a higher proportion of misclassified sensitive documents. From manually inspecting these documents, we see misclassifications of purely personal information within emails with little surrounding context. One often misclassified purely personal message is an email that is sent as a test. This email does not expose sensitive personal information, but is also not work-related, hence is treated as purely personal. Verbose responses show the keyword 'Enron' in the message creates a relationship to work. Therefore, our prompting strategy stating personal messages are not related to work ($S\_EC$) is not useful here. Within a larger collection, few-shot examples that are similar to these test messages may be useful for correctly classifying this type of message as personal.

The majority of misclassified sensitive emails are related to work. From our analysis a distinctly misclassified sensitive document containing the statement 'I heard you were a big hit,' within an email thread, should be treated as personal but in a professional context is not correctly classified. We are interested in exploring if these LLMs could utilise their pre-trained understanding and follow our instructions for sensitivity classification. Therefore, to assist zero-shot learning, we actively interact with the LLMs, evaluating if we can improve sensitivity classification for this document. Mistral can successfully explain that this statement is a form of sentiment towards a colleague: *"The sentence 'I heard you were a big hit' is an expression often used to convey that the person being referred to has been very successful or popular in a particular situation. The phrase 'a big hit' means that the person or thing in question has been well-received and has gained a lot of approval or admiration from others."*. We further embed this response as a hint within our system prompt; however, the classification is still incorrect with a verbose response explaining *"It does not contain any sensitive personal information. The phrase 'I heard you were a big hit is' a compliment and does not reveal any personal information"*. Despite understanding this phrase is a comment about a person, which aligns with our $S\_EC$ system context, the model still incorrectly classifies the message as non-sensitive. This non-automated approach demonstrates generative LLMs capabilities at describing the content within documents which may help a user identify the sensitive information; however, difficulty following our defined instructions limits the performance for automatically identifying sensitive information.

Overall, we find from exploring the most misclassified documents demonstrates that pre-trained LLMs do not always fully understand how to interpret sensitive personal information that is within the professional context of email documents. We believe further exploration of instructions to interpret information about sensitive personal information would be useful for identifying this misclassified sensitive information.

# 7. CONCLUSIONS

In this paper, we proposed novel prompt engineering strategies to improve the zero-shot performance of open-source generative LLMs in sensitivity classification. We use instructions to influence LLMs understanding of sensitive personal information within emails by introducing context about sensitive and non-sensitive email categories, and a personal sensitivity definition ($S\_EC$, $NS\_EC$, $PS$), few-shot prompting ($FS$), and chain-of-thought prompting ($CoT$). Our experiments on the SARA dataset found that basic classification prompting strategies (*Base*) were not sufficient at classifying sensitive personal information. We found using prompt engineering did improve the effectiveness of classifying sensitive personal information with generative LLMs. All of our prompting strategies introducing context within the system prompt made some significant improvements (according to McNemar's test) compared to our *Base* prompt strategy. Furthermore, we found that few-shot prompting made significant improvements for Mistral; however, our proposed $CoT$ strategy was ineffective. The best performing model and prompt was Mistral with context about sensitive email categories, sensitive personal information laws and few-shot examples ($S\_EC + PS + FS$). Furthermore, Mistral-$S\_EC + PS + FS$ as a sensitivity classifier performed competitively against existing machine learning strategies. We conclude generative LLMs do not outperform existing strategies; however, they also do not require training data. We believe this zero-shot learning capability, enhanced by prompts, will be useful in adapting to and managing new sensitivities. Zero-shot models can also be ethically advantageous because we do not require collection of sensitive data for training.

## 7.1 Future Work

From our evaluation, we know our prompting strategies do not fully cover every context required to identify all sensitive personal information. As well as this, refining these prompts to include appropriate context can be time-consuming and difficult. A new framework, DSPy [22], claims to replace manual prompt engineering using auto-tuned prompts. It could be useful to investigate if an optimised basic instruction prompt was more effective at sensitivity classification than our manually engineered prompts. It would also be useful to explore if using our manually engineered instructions as a starting point for prompt optimisation produces more effective prompts for identifying sensitive information. Furthermore, some models did not respond well to our few-shot examples, and DSPy is able to generate synthetic examples which may mitigate the confusion models had in our experiments when using few-shot prompting.

These instruction-based generative LLMs can be used to answer queries. Recent exploration of sensitivity-aware search [34] aims to identify relevant and non-sensitive documents. These LLMs could be prompted with appropriate system instructions to both protect sensitive information and score relevant documents given a query.

**Acknowledgments.** I would like to thank my supervisor, Dr. Graham McDonald, for his valuable comments and advice throughout the year.

## 8. REFERENCES

[1] D. Adiwardana, M.-T. Luong, D. R. So, J. Hall, N. Fiedel, R. Thoppilan, Z. Yang, A. Kulshreshtha, G. Nemade, Y. Lu, et al. Towards a human-like open-domain chatbot. *arXiv preprint arXiv:2001.09977*, 2020.

[2] M. M. Amin, E. Cambria, and B. W. Schuller. Will affective computing emerge from foundation models and general ai? a first evaluation on chatgpt. *arXiv preprint arXiv:2303.03186*, 2023.

[3] T. N. Archives. The digital landscape in government 2014-15. `https://www.nationalarchives.gov.uk/information-management/manage-information/policy-process/reviewing-digital-records-management-government/research/`, 2016. Last accessed: 2023-12-15.

[4] J. R. Baron, N. W. Rollings, and D. W. Oard. Using chatgpt for the foia exemption 5 deliberative process privilege. In *Proceedings of the Third International Workshop on Artificial Intelligence and Intelligent Assistance for Legal Professionals in the Digital Workplace (LegalAIIA 2023) co-located with the 19th International Conference on Artificial Intelligence and Law (ICAIL 2023), Braga, Portugal, June 19, 2023*, volume 3423 of *CEUR Workshop Proceedings*, pages 32–48. CEUR-WS.org, 2023.

[5] J. R. Baron, M. F. Sayed, and D. W. Oard. Providing more efficient access to government records: a use case involving application of machine learning to improve foia review for the deliberative process privilege. *ACM Journal on Computing and Cultural Heritage (JOCCH)*, 15(1):1–19, 2022.

[6] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[8] H. Fei, B. Li, Q. Liu, L. Bing, F. Li, and T.-S. Chua. Reasoning implicit sentiment with chain-of-thought prompting. *arXiv preprint arXiv:2305.11255*, 2023.

[9] O. for National Statistics. Freedom of information statistics: annual 2022 bulletin. `https://www.gov.uk/government/statistics/freedom-of-information-statistics-annual-2022/freedom-of-information-statistics-annual-2022-bulletin`, May 2023. Last accessed: 2023-12-15.

[10] E. Frantar, S. Ashkboos, T. Hoefler, and D. Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*, 2022.

[11] G. Gambarelli, A. Gangemi, and R. Tripodi. Is your model sensitive? spedac: A new resource for the automatic classification of sensitive personal data. *IEEE Access*, 11:10864–10880, 2023.

[12] T. Gao, A. Fisch, and D. Chen. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*, 2020.

[13] T. Gollins, G. McDonald, C. Macdonald, and I. Ounis. On using information retrieval for the selection and sensitivity review of digital public records. In *PIR@SIGIR*, pages 39–40, 2014.

[14] S. Government. Average response times for an foi request: Foi release. `https://www.gov.scot/publications/average-response-times-for-an-foi-request-foi-release/`, May 2023. Last accessed: 2023-12-15.

[15] X. Han, W. Zhao, N. Ding, Z. Liu, and M. Sun. Ptr: Prompt tuning with rules for text classification. *AI Open*, 3:182–192, 2022.

[16] P. He, X. Liu, J. Gao, and W. Chen. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*, 2020.

[17] M. A. Hearst. Teaching applied natural language processing: Triumphs and tribulations. In *Proceedings of the Second ACL Workshop on Effective Tools and Methodologies for Teaching NLP and CL*, pages 1–8, 2005.

[18] M. Iqbal, K. Shilton, M. F. Sayed, D. Oard, J. L. Rivera, and W. Cox. Search with discretion: Value sensitive design of training data for information retrieval. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW1):1–20, 2021.

[19] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[20] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[21] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. d. l. Casas, E. B. Hanna, F. Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

[22] O. Khattab, A. Singhvi, P. Maheshwari, Z. Zhang, K. Santhanam, S. Vardhamanan, S. Haq, A. Sharma, T. T. Joshi, H. Moazam, et al. Dspy: Compiling declarative language model calls into self-improving pipelines. *arXiv preprint arXiv:2310.03714*, 2023.

[23] B. Klimt and Y. Yang. The enron corpus: A new dataset for email classification research. In *European conference on machine learning*, pages 217–226. Springer, 2004.

[24] J. Kocoń, I. Cichecki, O. Kaszyca, M. Kochanek, D. Szydło, J. Baran, J. Bielaniewicz, M. Gruza, A. Janz, K. Kanclerz, et al. Chatgpt: Jack of all trades, master of none. *Information Fusion*, page 101861, 2023.

[25] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

[26] J. O. Krugmann and J. Hartmann. Sentiment analysis in the age of generative ai. *Customer Needs and Solutions*, 11(1):1–19, 2024.

[27] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

[28] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen,

O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[29] S. MacAvaney, A. Yates, S. Feldman, D. Downey, A. Cohan, and N. Goharian. Simplified data wrangling with ir_datasets. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2429–2436, 2021.

[30] G. McDonald, C. Macdonald, and I. Ounis. Using part-of-speech n-grams for sensitive-text classification. In *Proceedings of the 2015 International conference on the theory of information retrieval*, pages 381–384, 2015.

[31] G. McDonald, C. Macdonald, and I. Ounis. Enhancing sensitivity classification with semantic features using word embeddings. In *Advances in Information Retrieval: 39th European Conference on IR Research, ECIR 2017, Aberdeen, UK, April 8-13, 2017, Proceedings 39*, pages 450–463. Springer, 2017.

[32] G. Mcdonald, C. Macdonald, and I. Ounis. How the accuracy and confidence of sensitivity classification affects digital sensitivity review. *ACM Transactions on Information Systems (TOIS)*, 39(1):1–34, 2020.

[33] G. McDonald, C. Macdonald, I. Ounis, and T. Gollins. Towards a classifier for digital sensitivity review. In *Advances in Information Retrieval: 36th European Conference on IR Research, ECIR 2014, Amsterdam, The Netherlands, April 13-16, 2014. Proceedings 36*, pages 500–506. Springer, 2014.

[34] G. McDonald and D. Oard. Search among sensitive content. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28-April 1, 2021, Proceedings*, page 730. Springer, 2021.

[35] J. McKechnie and G. McDonald. Sara: A collection of sensitivity-aware relevance assessments. *arXiv preprint arXiv:2401.05144*, 2024.

[36] Q. McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

[37] H. Narvala, G. McDonald, and I. Ounis. Sensitivity review of large collections by identifying and prioritising coherent documents groups. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 4931–4935, 2022.

[38] H. Naveed, A. U. Khan, S. Qiu, M. Saqib, S. Anwar, M. Usman, N. Barnes, and A. Mian. A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*, 2023.

[39] J. Noh and R. Kavuluru. Improved biomedical word embeddings in the transformer era. *Journal of biomedical informatics*, 120:103867, 2021.

[40] T. F. on Technical Approaches for Email Archives. The future of email archives. `https://www.clir.org/pubs/reports/pub175/`, August 2018. Last accessed: 2023-12-15.

[41] R. OpenAI. Gpt-4 technical report. *arXiv*, pages 2303–08774, 2023.

[42] D. W. Otter, J. R. Medina, and J. K. Kalita. A survey of the usages of deep learning for natural language processing. *IEEE transactions on neural networks and learning systems*, 32(2):604–624, 2020.

[43] U. Parliament. Freedom of information act 2000. `https://www.legislation.gov.uk/ukpga/2000/36/contents`, 2000. Last accessed: 2023-12-15.

[44] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[45] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[46] D. S. Pereira, F. Falcão, L. Costa, B. S. Lunn, J. M. Pêgo, and P. Costa. Here's to the future: Conversational agents in higher education-a scoping review. *International Journal of Educational Research*, 122:102233, 2023.

[47] R. Puri and B. Catanzaro. Zero-shot text classification with generative language models. *arXiv preprint arXiv:1912.10165*, 2019.

[48] C. Qin, A. Zhang, Z. Zhang, J. Chen, M. Yasunaga, and D. Yang. Is chatgpt a general-purpose natural language processing task solver? *arXiv preprint arXiv:2302.06476*, 2023.

[49] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897, 2020.

[50] R. Rehurek and P. Sojka. Gensim–python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, 3(2), 2011.

[51] S. Roller, E. Dinan, N. Goyal, D. Ju, M. Williamson, Y. Liu, J. Xu, M. Ott, K. Shuster, E. M. Smith, et al. Recipes for building an open-domain chatbot. *arXiv preprint arXiv:2004.13637*, 2020.

[52] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

[53] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[54] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

[55] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

[56] X. Wei, X. Cui, N. Cheng, X. Wang, X. Zhang, S. Huang, P. Xie, J. Xu, Y. Chen, M. Zhang, et al. Zero-shot information extraction via chatting with chatgpt. *arXiv preprint arXiv:2302.10205*, 2023.

[57] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, and D. C. Schmidt. A prompt pattern catalog to enhance prompt engineering with chatgpt. *arXiv preprint arXiv:2302.11382*, 2023.

[58] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, 10 2020. Association for Computational Linguistics.

[59] W. Yin, J. Hay, and D. Roth. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. *arXiv preprint arXiv:1909.00161*, 2019.

[60] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

[61] Y. Zhou, A. I. Muresanu, Z. Han, K. Paster, S. Pitis, H. Chan, and J. Ba. Large language models are human-level prompt engineers. *arXiv preprint arXiv:2211.01910*, 2022.

# 9. APPENDIX

**Table 6: Mistral and Llama 2 have been queried and demonstrate their understanding of sensitive personal information.**

| |
|---|
| **Prompt**<br>[INST] What does sensitive personal information mean? [/INST] |
| **Mistral answer**<br>Sensitive personal information refers to data that is highly personal or sensitive in nature, and is typically protected by laws and regulations that govern the collection, use, and disclosure of personal data. Here are some examples of sensitive personal information:<br>1. Financial information: This includes credit card numbers, bank account numbers, and other financial information that can be used to access an individual's financial accounts.<br>2. Health information: This includes medical records, health insurance information, and other personal health data that is protected by the Health Insurance Portability and Accountability Act (HIPAA) in the United States.<br>3. Biometric information: This includes biometric data such as fingerprints, facial recognition data, and other unique identifiers that can be used to authenticate an individual's identity.<br>4. Personal identity documents: This includes passports, driver's licenses, and other government-issued identification documents that can be used to verify an individual's identity.<br>5. Location data: This includes location information from GPS-enabled devices, mobile phones, and other sources that can be used to track an individual's location and movements.<br>6. Children's information: This includes personal information about children, such as their names, addresses, and birthdates, that is protected by the Children's Online Privacy Protection Act (COPPA) in the United States.<br>7. Sexual orientation or gender identity: This includes personal information about an individual's sexual orientation or gender identity, which is protected by various laws and regulations around the world.<br>8. Racial or ethnic information: This includes personal information about an individual's race or ethnicity, which is protected by various laws and regulations around the world.<br>9. Political beliefs or activities: This includes personal information about an individual's political beliefs or activities. |
| **Llama 2 answer**<br>Sensitive personal information refers to data that is highly confidential and personal to an individual. This type of information is considered sensitive because its disclosure could result in harm, embarrassment, or invasion of privacy. Some common examples of sensitive personal information include:<br>1. Social Security numbers<br>2. Financial account information, such as credit card or bank account numbers<br>3. Health information, including medical records and conditions<br>4. Biometric data, such as fingerprints or facial recognition information<br>5. Passwords or other authentication information<br>6. Personal identification numbers (PINs)<br>7. Home addresses or other contact information<br>8. Racial or ethnic origin<br>9. Religious or philosophical beliefs<br>10. Sexual orientation or gender identity<br><br>It's important to protect sensitive personal information from unauthorized access, use, or disclosure. |

**Table 7: Mistral and Llama 2 have been queried and demonstrate their understanding of FOIA Section 40 as a reason for needing to protect sensitive personal information.**

**Prompt**

[INST] Describe UK FOIA Section 40. [/INST]

**Mistral answer**

The United Kingdom’s Freedom of Information Act (FOIA) contains several exemptions that allow public authorities to withhold certain information from disclosure. One such exemption is Section 40, which relates to personal information.

Section 40(1) of the FOIA provides that information is exempt if its disclosure would breach one of the data protection principles, as set out in the Data Protection Act 1998 (DPA). These principles include requirements that personal data be obtained and processed fairly and lawfully, be accurate and kept up to date, and not be disclosed without the consent of the data subject, unless one of the specified conditions is met.

Section 40(2) of the FOIA provides an additional exemption for personal data that is held by public authorities for the purposes of preventing or detecting crime. In this case, disclosure of the information would be exempt if it would be likely to prejudice the prevention or detection of crime, or the apprehension or prosecution of offenders.

Section 40(3) of the FOIA provides an exemption for personal data that is held by public authorities for the purposes of safeguarding the physical or mental health of an individual. In this case, disclosure of the information would be exempt if it would be likely to cause serious harm to the physical or mental health of the individual, or to someone else, if disclosed.

Section 40(4) of the FOIA provides an exemption for personal data that is held by public authorities for the purposes of assessing or collecting tax or duty. In this case, disclosure of the information would be exempt if it would be likely to prejudice the assessment or collection of tax or duty.

It is important to note that the exemptions in Section 40 are not absolute.

**Llama 2 answer**

Section 40 of the Freedom of Information Act (FOIA) in the United Kingdom provides an exemption for certain types of information that are considered to be sensitive or confidential. This exemption is known as ”section 40(2)” or ”section 40(3)” depending on the specific circumstances

Section 40(2) applies when the information requested is held by a public authority, but is not available for release because it is protected by one of the following exemptions:

- Exemption 1: Information that is protected by legal professional privilege (LPP)
- Exemption 2: Information that is confidential or privileged in the context of legal proceedings
- Exemption 3: Information that is confidential or privileged in the context of a contract or other agreement
- Exemption 4: Information that is confidential or sensitive in the context of a trade secret or commercial confidentiality

Section 40(3) applies when the information requested is held by a public authority, but is not available for release because it is protected by one of the following exemptions:

- Exemption 5: Information that is personal data and is protected under the Data Protection Act 2018
- Exemption 6: Information that is sensitive personal information and is protected under the Data Protection Act 2018
- Exemption 7: Information that is protected by the right to privacy and confidentiality under the Human Rights Act 1998

In both cases, the public authority is not required to confirm or deny the existence of the information, but must instead issue a notice stating that the information is exempt from disclosure under section 40. The notice must also explain why the information is exempt and provide details of any relevant advice or guidance that is available.