UNIVERSITY
OF OSLO

Master's thesis

Generation and Selection of Replacement Choices for Text Sanitization

Annika Willoch Olstad

Informatics: Language Technology
60 ECTS study points

Department of Informatics
Faculty of Mathematics and Natural Sciences

Spring 2023

Annika Willoch Olstad

Generation and Selection of Replacement Choices for Text Sanitization

Abstract

The right to privacy is a fundamental human right. This includes our right to protect and control our personal information. However, such information is present all around us, among others in text documents. Text sanitization techniques aim to mask text spans in documents holding such information, so that the text no longer identifies any individuals.

A common problem with most sanitization techniques is that they tend to completely remove personal information from the text document, thus making it harder to read, re-use or process for other purposes. Some approaches also replace such spans with other values that might alter the ground truth of the span and as a result of the document itself.

In this thesis, we address this issue by utilizing generalization for text sanitization. The objective is to sanitize text balancing both data privacy and data utility. Our approach consists of two steps. We first generate and suggest possible replacements for already detected *Personally Identifiable Information* (PII) spans that need to be masked. The replacements are generated using a combination of an ontology and rules, depending on each PII's semantic type. Then we use a machine learning model to choose the best replacement for a given span out of the suggestions.

To evaluate our approach, we extend an existing dataset for text sanitization with replacement choices selected by human annotators. The resulting dataset, named *WikiReplace*, is employed to assess the empirical validity of our replacement selection model. We find that our proposed approach is able to limit the use of deletion in text sanitization - resulting in more useful text documents with reduced privacy risk.

Acknowledgements

Writing this thesis has been a great learning experience, and I am grateful to everyone who supported me along the way.

First of all, I would like to thank Anthi Papadopoulou for her supervision during this master project. I appreciated every discussion and meeting we had, and I am immensely grateful for your generous sharing of knowledge. I would also like to thank my supervisor Pierre Lison for all his ideas and invaluable feedback on this work. Thank you both so much.

An enormous thank you, also, to all my friends and family having encouraged me throughout my thesis. A special thank you to Daniel for so generously supporting me, even though things ended up differently than we had imagined. An equally great thank you to my aunt Wenche for your endless hospitality and generosity.

Thank you all so much - this thesis would not have been possible without you.

Contents

List of Tables

List of Figures

Listings

Chapter 1

Introduction

In this thesis, we explore the task of *generalization* as a tool for *text sanitization*. Text sanitization refers to the task of editing documents so that personal information in the text is masked and the identity of individuals referenced in the document are protected. An alternative to removing such spans is to generalize them to mask personal information, by making a term more general so that it can still be useful and informative without introducing additional privacy risk. For instance, the text span *"Oslo"* can be generalized to the less risky formulation *"city in Norway"*. This generalization is less specific than *"capital of Norway"* for example, but still more informative than just *"city"* or even *"city in Europe"*.

As part of a text sanitization pipeline, the goal of generalization is to generalize personal information in documents so that no individual can be identified in said document, while also maintaining as much of the readability and utility of the resulting textual data as possible. This, so that they can be used for secondary purposes and analyses.

In this thesis, we explore how to create an ontology that can be used to propose possible hierarchical text replacements for a given span, and how to use that ontology to set up an annotation effort for the task of text replacement selection using human annotators. Within the scope of the thesis is also how to use the resulting annotated dataset to train a machine learning model to automatically select the best replacement out of these suggestions for a given document, and lastly, how to evaluate such a system, both in regards of *data privacy* and *data utility*.

1.1 Motivation

In today's digital society, the amount of available data is continuously growing. Every single day, we leave behind new traces of information, whether it being paying with a bank card or posting on social media. Much of this produced data contains personal data, also known as *Personally Identifiable Information (PII)*. This is information that can be used to identify individuals, for instance names, ID-numbers or someone's gender either on their own or combined with other types of PII (Domingo-Ferrer et al., 2016). Along with the growth in both

amount and availability of data, the importance of *data privacy* has accordingly increased.

The right to privacy is a fundamental human right, as stipulated by Article 12 of the Universal Declaration of Human Rights Declaration. Data protection is an extension of this right - aiming to protect and control the distribution of personal data, for all individuals. To ensure the adherence to data privacy, these rights have been incorporated in various legal frameworks, such as the General Data Protection Regulation (GDPR) (GDPR, 2016) in Europe or the Health Insurance Portability and Accountability Act (HIPAA) (HIPAA, 1996) in the U.S.A..

Much of the data containing PIIs is textual and comes in an unstructured format, such as tweets, articles, clinical notes, customer service conversations, and more. Despite being unstructured, it can be of immense value for both scientific and commercial purposes. In academia, data is the basis of various scientific studies, e.g. as training data for machine learning (ML) models. The knowledge and experience from such studies can also be used, among others, for domains such as the legal or medical domain where court cases or electronic health records can be safely shared with third parties or used for secondary statistical analyses or training purposes. When meeting these demands for data availability, it is important to ensure that the individual's right to data protection is respected. In order to fulfill both needs, we must *sanitize* the data, so that no individual can be identified from the text used.

However, simply deleting detected PII spans may result in large semantic loss of the data. It may remove unnecessary information or even change the truth value of the text. The result is data with low readability and low utility. Meeting the demands of (re)usable data, while respecting the laws and rights of privacy, is thus a complex task. In this thesis we explore how we can use generalization for this purpose.

1.2 Objectives

The main objective of this thesis is to establish a method for generalizing detected PII spans in text, in a way that preserves the readability and data utility of the text without introducing any additional privacy risk. The modelling objective is thus to create a system that can propose generalizations for each identified PII, and select a replacement among these, as seen in Figure 1.1. A sub-objective of this is to create a proper dataset that can be used for machine learning purposes for automatic generalization in text sanitization.

1.2.1 Research questions

In this thesis we aim to answer the following research questions:

> **RQ1:** How can we create an ontology that suggests appropriate hierarchical replacements for different types of PII?
>
> **RQ2:** Subsequently, how can we use this ontology to suggest replacements and set up an annotation task to manually annotate and release a dataset of possible text replacement choices?

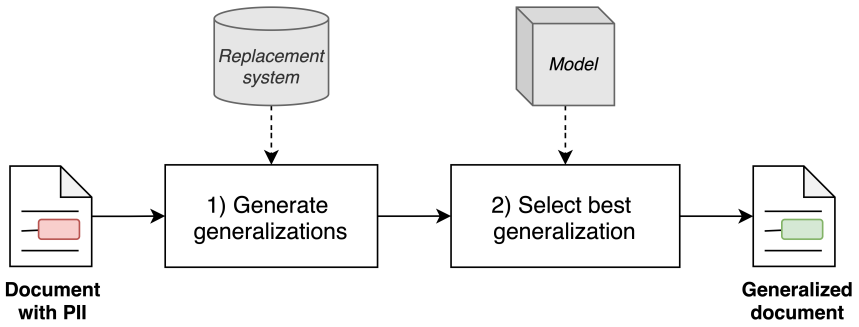

Figure 1.1: Overview of modelling objective, where the approach first generates possible replacements and then selects the best one

RQ3: How can we use this data to choose and train a model to automatically select the best replacement for a token, without increasing the privacy risk, yet keeping a high data utility?

RQ4: How can we evaluate such task both in regards to privacy risk, but also the resulting data utility of the text?

1.3 Thesis outline

This thesis contains 6 chapters.

Chapter 2: in this chapter we discuss the task of generalization as text sanitization, and present essential terminology. We also present previous, relevant work on this topic, including approaches from various scientific fields. Finally, we introduce the dataset used in this thesis.

Chapter 3: in this chapter we present our approach to creating an ontology for text generalization, including the knowledge base (Wikidata) used for this work.

Chapter 4: in this chapter we present the WikiReplace dataset - a human-annotated dataset for generalization. We also present the annotation effort resulting in this dataset.

Chapter 5: in this chapter we utilize the annotations from the WikiReplace dataset to train machine learning models for automatic selection of generalization options.

Chapter 6: in this chapter we summarize the work conducted in this thesis, and conclude on our research questions. We also discuss possible future work.

Chapter 2

Background

In this chapter, we provide the base knowledge needed for our work in the coming chapters. We define and discuss approaches and terminology for text sanitization, generalization and ontology development. We also introduce previous work from different fields relevant for the research in this thesis.

2.1 Text Sanitization

Text sanitization is a technique that aims to protect the identity of individuals in text, with the goal of enhancing the level of privacy protection. This process includes identifying and removing or modifying PII that may lead to the identification of an individual (Papadopoulou, Yu, et al., 2022).

PII can be divided into two main categories (Elliot et al., 2016):

Direct identifiers Any information that directly discloses the identity of an individual, such as full name, social security number, biometric records, and more.

Quasi identifiers Information that in isolation does not identify an individual, but when combined with other identifiers may disclose the identity of said individual. Postal codes, gender, religion, political beliefs, or sexual orientation are examples of such identifiers. For example, Sweeney (2000) showed that with using only the combination of gender, birth date and postal code information, one could identify between 63% and 87% of the U.S. population.

The collection and processing of any data belonging to EU citizens and residents that contains such identifiers is subject to the GDPR, but when the data no longer can lead to the identification of an individual, it is said to be anonymous and it falls outside the scope of this legislation (cf. Recital 26 GDPR) (Weitzenboeck et al., 2022). There are various approaches for masking such PII, some of which are described in Section 2.4. The approach we present in this thesis is that of replacing the PII with a more general term instead of removing spans, so that the semantic loss of the data is kept to a minimum while also being conscious of the privacy risk introduced.

2.1.0.1 Terminological Note on Text Anonymization

Data privacy frameworks like the GDPR mandates a need of complete *anonymization* of data, in order for it to be collected and processed without the explicit and informed consent of the affected individuals. This anonymization process must be both complete and irreversible (Lison et al., 2021). *All* identifiers must thus be removed or masked, and it must not be possible to reverse this process to obtain the original, personal information. In practice, obtaining such full anonymization has been shown to be nearly impossible (Weitzenboeck et al., 2022), unless the original data has been deleted.

In the literature, the task of removing personal identifiers from textual data has also been referred to with various names such as the task of *de-identification* (Carrell et al., 2013) or *pseudonymization* (Dalianis, 2019), in addition to sanitization and anonymization, among others.

However, to avoid the vagueness of the actual requirements for data to be considered "anonymized" and to differentiate our approach from applications for structured data, we will use the term text sanitization in this thesis, with the definition provided earlier in this section (2.1).

2.2 Generalization

Generalization is a text sanitization technique, which replaces PII with less specific terms. This is typically done for quasi identifiers, as direct identifiers, such as social security or passport numbers, are nearly impossible to generalize appropriately and should thus be removed. An example is shown below: *"Oslo"* can be generalized with the more general phrasing *"city in Norway"*. Using this technique preserves the semantic content and truth value of the expression, while making it more difficult to use for re-identification of individuals. This does, however, assume that the generalization is performed in a manner that truly decreases the privacy risk. Replacing *"Oslo"* with *"capital of Norway"* is for instance not a good generalization. There is only one capital in Norway, and the privacy risk thus remains at the same level as for the original text span.

Original Text
She lives in **Oslo**.

Generalized text

1. She lives in a **[city in Norway]**.
2. *She lives in the **[capital of Norway]**.

The same problem can be seen in the following example:

Original Text
She studies at the **the University of Oslo**.

Generalized text

1. She studies at a **[university in Norway]**.
2. *She studies at the **[oldest university in Norway]**.

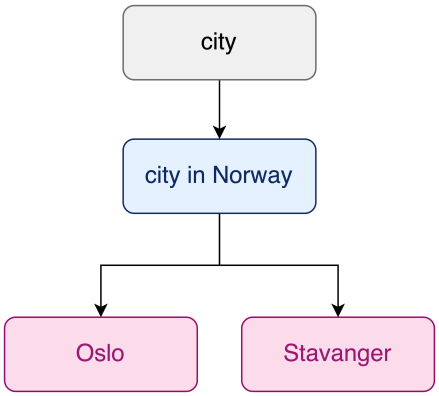

Figure 2.1: Example of a simple ontology with only "is a"-relations between concepts.

Here, the most appropriate generalization of *"University of Oslo"* is *" university in Norway"*, not *"oldest university in Norway"*. The latter is avoided in this case as it is just as specific as *"University of Oslo"* - it is known and easily verifiable that the University of Oslo is in fact the *"oldest university in Norway"*.

As described above, in the terminological note on anonymization (Subsection 2.1.0.1), guaranteeing anonymization is hard to nearly impossible (Weitzenboeck et al., 2022). In *Opinion 05/2014 on Anonymisation Techniques in Article 29 Working Party*, a former data privacy advisory group of the European Commission, the task of generalization is similarly described as rather challenging, since it *"[...] does not allow effective anonymization in all cases"* (Party, 2014, p.16). In this thesis, we target some of the challenges of the generalization technique, in regards to data protection.

2.3 Ontology

An ontology is a formal representation of knowledge, aiming to model both the isolated knowledge about entities and the relations between them. The resulting representation can be visualized as a graph, where the nodes are the different concepts present in the world and the edges are the relations between them. An example of such a knowledge representation is provided in Figure 2.1. The graph in this figure shows an ontology in one of its simplest form. The concepts (*"city"*, *"city in Norway"*, *"Oslo"* and *"Stavanger"*) are related with a simple *"is a"*-relation, given as follows:

1. *"Oslo"* **is a** *"city in Norway"*
2. *"Stavanger"* **is a** *"city in Norway"*
3. every *"city in Norway"* **is a** *"city"*

The result is a hierarchical structure, a *taxonomy*, where the top term is the most general, and the lower nodes are more specific.

Ontologies are thus a way to structure data. Consequently, they have many use cases, e.g. allowing the sharing of knowledge across different domains and languages. In this work, we will use ontologies for generalization, by utilizing the hierarchical structure described above and shown in Figure 2.1.

There are languages developed for the sole purpose of creating ontologies, such as the logic-based language *Web Ontology Language (OWL)*[1]. However, in this work we propose an alternative approach, mainly based on using lists in Python. Traditional ontology languages, such as OWL, are as a consequence omitted in this thesis. This decision is justified and further explained in Chapter 3.

2.3.1 Ontology Base

There are various ways to construct ontologies. In this thesis we base our ontology on a knowledge base (KB), i.e. a base containing structured information - knowledge. More specifically, we use Wikidata[2], as it has been shown to be a useful semantic framework when forming the basis for an ontology, e.g. a medical database (Turki et al., 2019). Similarly to this study, we aim to create a machine-and human-readable database for a more generalized domain, where entities are linked to their respective generalizations.

2.3.1.1 Wikidata

On the main page of its website, Wikidata is described as *"a free and open knowledge base that can be read and edited by both humans and machines"*.[3] This KB contains *items* linked together with *properties*. Items are the pieces of information and properties are the relations between them. Elements in both categories have a dedicated *identifier* (ID), distinguishing them from other elements in Wikidata. Many elements also have a *label* (name) and *aliases* (variations of labels), making it more readable for humans. In addition, most items also have other pieces of information, such as the name in various languages, description, country of citizenship or occupation.

An excerpt of an example of a Wikidata item is provided in Figure 2.2. We observe that the item, among others, has the label *"Oslo"*, the ID *"Q585"*, and the aliases *"Christiana", "Kristiania", "NOOSL"* and *"Oslo, Norway"*. This particular item also has the *"instance of"* property in its *Statements*-section, containing various related values, i.e. other items that *"Oslo"* is an instance of.

In this thesis we utilize the Statements-section of a Wikidata entity, which is where the majority of the information lies. In particular, we focus on the taxonomic relations expressed in specific membership properties, namely *instance of* (P31), *subclass of* (P279), *is metaclass of* (P8225) and *part of* (P361). For example, as seen in Figure 2.2, the words *"capital city", "big city"*, and *"administrative centre"* are some of the items related to the item *"Oslo"* with the "instance of" property. A further description and discussion on how we use Wikidata for the construction of our ontology is provided in Chapter 3.

[1] https://www.w3.org/OWL/
[2] https://www.wikidata.org/
[3] See footnote 2

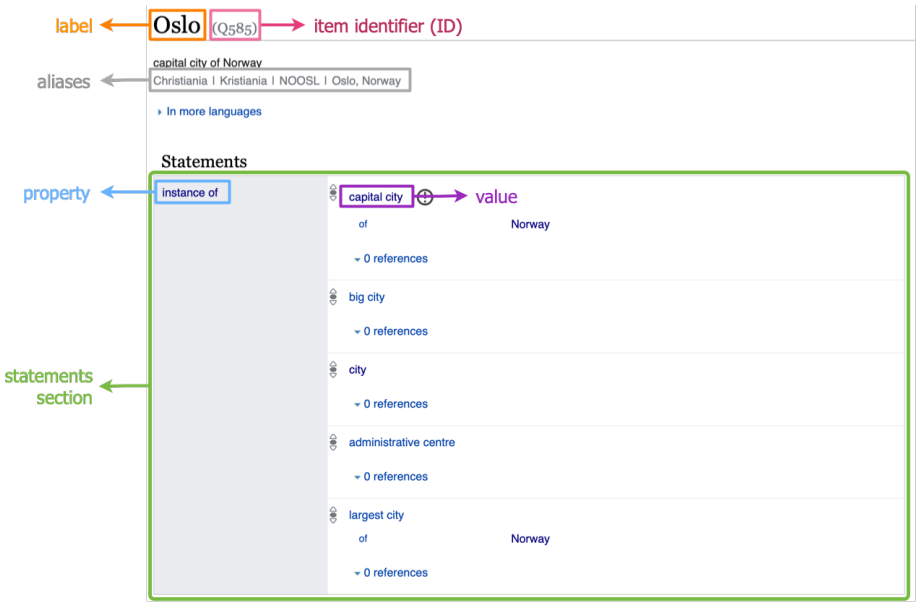

Figure 2.2: Example of Wikidata item: "Oslo"

2.4 Previous Work

This section provides a detailed account of previous work relevant to this thesis. In particular, it elaborates on the various existing approaches for the task of text sanitization and generalization, which can be divided into two categories: *Natural Language Processing (NLP)* and *Privacy Preserving Data Publishing (PPDP)* approaches.

2.4.1 NLP Approaches

In the field of Natural Language Processing (NLP), there have been various attempts to sanitize, and even anonymize, data. Most of the work has focused on the task of *de-identification*, where predefined categories of PII are either removed or masked with a black box or *** in the dataset (Lison et al., 2021). Two NLP-based solutions dealing with already detected PII spans are *pseudonymization* and *lexical substitution*, described below in the subsections 2.4.1.1 and 2.4.1.2 respectively.

In general, most of the NLP approaches share the advantage of being applicable to unstructured text data, as they consider the linguistic traits and relations between words. The semantics of a PII and that of its suggested replacement thus remains very similar. The main challenge for the NLP approaches is that the majority focuses on identifying a pre-defined set of identifiers. As a consequence, they focus less on the privacy perspective related to the task of generalization. More specifically, they focus on replacing PII in isolation, rather than considering how they can be combined with other information to identify a person.

2.4.1.1 Pseudonymization

Pseudonymization is an approach where the identifier is replaced by a pseudonym (Dalianis, 2019). Several techniques have been proposed for this, some of which are the *Hiding in Plain Sight (HIPS)* method (Carrell et al., 2013) and random replacement of names (Dalianis, 2019).

The HIPS method replaces all identified sensitive information with synthetic surrogates (Carrell et al., 2013). According to Carrell et al. (2013), this obfuscation allows for some unidentified PII spans, since it makes it difficult to distinguish the synthetic replacements from the original identifiers. However, as the PII spans are replaced with synthetic replacements, there are no guarantees for preserving the ground truth value of the expression. For instance, replacing the name of a person or the name of a city with a fake name, will alter the references in the text, and thus the expressed truth value.

Dalianis (2019) proposes a different method for pseudonymization: a rule-based approach where already tagged Protected Health Information (PHI) is replaced with surrogates. He considers eight types of PHI: names, phone numbers, dates, ages, healthcare units and other locations. These identifiers are replaced with random selections from various lists, e.g. listing streets in Stockholm, locations in Sweden and common first and last names. One of the main motivations for this approach is that the patient records would not contain unusual replacements taking the focus away from the medical content of the records, while still preventing re-identification. However, this method leads to the revelation of many of the pseudonymized patient records, precisely because of the unusual combinations of first names and surnames or the misalignment between family relationships and gender of names. An alternative, circumventing this drawback, is to replace the names with more generic identifiers such as *"Person 1"* or *"A"*, as is for instance done by Lovdata[4].

A variant of pseudonymization is also presented by Volodina et al. (2020). In this work, personal information is detected, labeled and replaced in essays written by Swedish learners. The authors propose a rule-based approach for the detection of PII, among other using regular expressions. The span replacements are extracted from external, public resources, such as GeoNames[5] and Swedish Central Statistics agency. Though the results are promising (successful identification of 89% of personal information), several remaining challenges are pointed out by the authors. This includes, but is not limited to, consistent problems with certain categories yielding both too many false positives, i.e. excessive detection of personal information, and too many false negatives, meaning that PII that should have been flagged are not detected (Volodina et al., 2020).

2.4.1.2 Lexical Substitution

An alternative to pseudonymization is lexical substitution. This method substitutes a target word with a similar lexical entity, e.g. a synonym or a hypernym, so that the semantic meaning of the expression remains as similar as possible (McCarthy & Navigli, 2009). To prevent re-identification it is important that the substitution is not a near identical synonym, since this will not make the identifier more general and may allow for re-identification. Some of the proposed

[4]https://lovdata.no
[5]https://www.geonames.org

solutions of lexical substitutions are BERT-based (Zhou et al., 2019) and with the use of other neural language models(LM) (Arefyev et al., 2020).

In the first approach, the authors utilize the language model BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) to propose and validate replacements of target words. They also show that using the context of a target word can improve the replacement suggestions provided by BERT.

The latter work, presented by Arefyev et al. (2020), compares various popular neural language- and masked language models (such as text2vec, ELMo and BERT) and their performance on the task of lexical substitution, both intrinsically end extrinsically. They note comparable results between simple unsupervised approaches utilizing neural language models and more traditional supervised approaches. Similarly to Zhou et al. (2019), their results also suggest that including information about the target terms improves the overall quality of the lexical substitution (Arefyev et al., 2020).

2.4.2 PPDP Approaches

Privacy Preserving Data Publishing approaches differ from NLP approaches by having a different focus. The main objective of PPDP approaches is to transform data prior to release to satisfy a formal privacy guarantee (Lison et al., 2021). PPDP models, thus, typically specify a privacy condition that needs to be fulfilled by the model before the data can be published. Generalization is one of several techniques that can fulfill this condition. Other, already established attempts to formalize such privacy models have been made, such as *k-anonymity* and *t-plausibility*. Furthermore there are techniques like *C-sanitization*, which, in addition to specifying a privacy model, also provide an implementation of the given model. These techniques are further described in the following subsections: 2.4.2.1, 2.4.2.2 and 2.4.2.3.

The PPDP approaches share the main advantage of being privacy oriented, ensuring that the possibility of identity disclosure is kept at a minimum. In addition, having a formal definition of a privacy model may also make it easier to evaluate, compared to the methods in 2.4.1. However, as opposed to NLP approaches, most PPDP solutions do not consider the linguistic challenges of unstructured text data. Among others, they disregard that a text is not a simple bag of words, but a document where words are linked to one another. As a consequence, PPDP models fail to solve the issues related to both the context in which identifiers appear in the data and how these entities can lead to re-identification through semantic inferences. They also fail to address the problem of detecting both identifiers and their appropriate substitution in a data set that is not structured, as pointed out by Lison et al. (2021).

2.4.2.1 k-anonymity

k-anonymity is a privacy model that addresses the problem of how the inference linking between quasi identifiers and other publicly available information, can lead to re-identification of an individual (Samarati & Sweeney, 1998). It introduces the notion of k-anonymity as a measure of how well protected the data is, in terms of data privacy. For a data to be released, k-anonymity requires that *"[...]every combination of values of quasi-identifiers can be instinctively matched*

to at least k individuals", as precised by Samarati and Sweeney (1998). This means that if the various combinations of PIIs are linked to k individuals or more, one can assume that the data will not lead to identity disclosure.

As most PPDP approaches, k-anonymity mainly refers to structured data. However, other works have tentatively adapted it for unstructured data, such as k-safety (Chakaravarthy et al., 2008) and k-confusability (Cumby & Ghani, 2011). Other approaches addressing the limitations of k-anonymity for unstructured data are t-plausibility (Anandan et al., 2012) and C-sanitization (Sánchez & Batet, 2016), both of which are described in the following.

2.4.2.2 t-plausibility

t-plausibility is a theoretic approach to text document privacy aiming to anonymize unstructured text data (Anandan et al., 2012). The intuition of the t-plausibility method is that *"[...]given a threshold t and an ontology, a sanitized text should be a plausible result of at least t base text documents."* (Anandan et al., 2012). The goal is thus replacing identifiers, using an ontology, so that the sanitized text is a reasonable result of at least t number of documents.

This work proposes several algorithms for the text sanitization of documents. The approaches are based on the traversing and pruning of hypernym trees for each identifier, extracted from WordNet (Miller, 1995). t-plausibility makes several assumptions, such as already having a method for the detection of PIIs and that each sensitive word is independent from others. The latter differs from the work we present in this thesis, as we do not make this assumption, but rather consider their relations. This is reflected in our annotation guidelines (see Subsection 4.3.1). In addition, we utilize a different knowledge base than WordNet, namely Wikidata.

2.4.2.3 C-sanitization

Another approach aiming to guarantee data protection is C-sanitization (Sánchez & Batet, 2016). As opposed to many of the other privacy focused approaches, this method does not only consider the replacement of identifiers, but also aims to avoid semantic inferences without removing all the semantics of the data. Their formally defined privacy model, named C-sanitization, thus balances both data privacy and data utility. This is done through an automatic sanitization process mimicking manual sanitization work where identifiers are replaced with suitable generalizations.

The approach assumes a domain knowledge K, representing the available knowledge that can be used for identity disclosure, e.g. web data. The goal is to generalize all PII so that no terms in a C-sanitized document disclose the identity of an individual, either in isolation or through semantic inferences. For each identifier in a text document, they retrieve generalizations from a knowledge base, in particular SNOMED-CT, WordNet and ODP. The suggestions are then iteratively evaluated, until a replacement fulfilling the specified privacy criteria is found. The strictness of the privacy level can be adjusted through an α parameter altering the trade-off between data protection and utility (Sánchez & Batet, 2016).

C-sanitization is thus a formally defined privacy model, avoiding the use of human annotators. This differs from what we present in this work, where

annotations by humans are used to create a dataset, which in turn is used to train a machine learning model to automatically select the most appropriate replacement. In addition, we consider a different knowledge base than the ones used in Sánchez and Batet (2016).

2.5 Summary

In summary, we have introduced and defined essential terms for text sanitization, such as personal identifiers, direct and quasi identifiers, anonymization, sanitization and generalization. We have also discussed strategies for ontology development, and knowledge base options, including how Wikidata has been seen to be a useful semantic framework when forming the basis of an ontology.

Furthermore, we have presented and discussed previous work of text sanitization, focusing on both NLP approaches, such as pseudonymization and lexical substitution, and PPDP approaches, including k-anonymity, t-plausibility and C-sanitization. The advantages and disadvantages of both approaches have also been discussed. We saw that the main limitation of NLP approaches is that they focus less on the privacy aspect of sanitization, whereas the PPDP approaches generally fail to consider the linguistic challenges of unstructured text data.

Chapter 3

Ontology

We design and implement a system for automatic generation of generalization suggestions for detected PII spans. To do this we utilize properties in Wikidata and heuristics, to create an ontology with generalizations for common terms. This is performed in three main steps:

1. creating the foundation of the ontology
2. further enhancing the ontology and
3. creating the surrounding system to use the ontology

In this chapter we present our approach to creating the ontology in further detail. In Section 3.1 we describe the initial construction of the ontology, including the Wikidata dump file and entities, the extraction of taxonomies from Wikidata and what properties we used. In Section 3.2 we identify possible improvements of the ontology and describe the implementation of these enhancements. Finally, in Section 3.3 we present the final ontology and how we utilize it for generalization.

3.1 Initial Design

This section presents the approach and technologies used when constructing the basis of the ontology, including the use of a knowledge base and tools for extracting taxonomies from this.

3.1.1 Knowledge Base

Wikidata is used as knowledge base in this work, forming the foundation of the ontology. Alternative information sources that could be used as basis of the ontology include the knowledge base DBpedia[1] and the lexical database WordNet (Miller, 1995). We choose Wikidata as it is more scalable with a larger coverage of relevant terms. For instance, a comparative study found that Wikidata returned both more appropriate data and more results than DBpedia, and that the former had more frequent additions and updates (Abián et al.,

[1] https://www.dbpedia.org

2018). In addition, we prefer Wikidata to a lexical database such as WordNet, due to the fact that in text sanitization we are likely to encounter many proper nouns (names of cities, organizations and more). A manual inspection indicates that Wikidata is much richer in this type of nouns than WordNet. For instance, *"Ada Lovelace", "Josephine"* and *"Larvik"* are all entities available in Wikidata, but not in the online version of WordNet[2]. We also want to avoid the risk of replacing a term with a synonym (i.e. other words in the same synsets in WordNet), as such a replacement may introduce a privacy risk, since a synonym is generally not sufficiently general compared to the original term.

3.1.1.1 Extracting Taxonomies from Wikidata

The ontology in this work will contain various terms and their corresponding lists of generalization suggestions. For instance, for the entry *"computer scientist"* the generalization list in the ontology may be the following:

computer scientist → scientist → person → individual

We observe here that *"scientist"* is more general than *"computer scientist"*, and that *"person"* in turn is more general than *"scientist"*. In other words *"scientist"* and *"person"* are hypernyms of *"computer scientist"*. The list thus forms a hierarchical structure, sorted from most specific to most general. As a consequence, the first generalization option will be the most informative one, but also the most revealing one, in regards to the possibility of identity disclosure. The last suggestion will be the least informative, and equally, the option introducing the least privacy risk. The generalization lists for other entities will also follow this ordering.

In most cases, there will be more than one generalization list for each term. Common for these are, as we see above, that they form a hierarchical structure between the terms, in regards to generality. In this work we utilize four properties defined in Wikidata, that in some way express such a hierarchical taxonomic relationship between the related entities (membership properties):

1. **P31** *instance of*: indicates that an entity A is an example of an entity B. [3]
2. **P279** *subclass of*: designates that an entity A is a type of another entity B, but not an instance of it.[4]
3. **P361** *part of*: describes the relation where an entity A is contained within another entity B.[5]
4. **P8225** *is metaclass for*: is used to denote the relation where all members of an entity A is a type of another entity B.[6]

Examples of each property are provided in Table 3.1. In the following, we use these properties to extract generalization lists for entities we add to the ontology.

[2] http://wordnetweb.princeton.edu/perl/webwn
[3] https://www.wikidata.org/wiki/Property:P31
[4] https://www.wikidata.org/wiki/Property:P279
[5] https://www.wikidata.org/wiki/Property:P361
[6] https://www.wikidata.org/wiki/Property:P8225

ID	Label	Example
P31	instance of	Atlantic Ocean *instance of* ocean
P279	subclass of	university student *subclass of* student
P8225	is metaclass for	hair type *is metaclass for* hair
P361	part of	knee *part of* leg

Table 3.1: Examples for each of the four Wikidata membership properties employed to construct the generalization ontology.

```
Oslo (Q585)
├─city (Q515)
│  └─classification of human settlements (Q98966309)
├─capital city (Q5119)
├─administrative centre (Q1306755)
├─big city (Q1549591)
│  └─classification of human settlements (Q98966309)
└─largest city (Q51929311)
```

Figure 3.1: Taxonomy for Wikidata entity "Oslo" (Q585) retrieved with Wikidata-Taxonomy-tool

To extract the taxonomy for entities according to the properties above, we use `Wikidata-Taxonomy`: a command line tool that can be used to extract hierarchical lists from Wikidata, according to a specified property[7]. For instance, we can use this tool to retrieve a list of generalizations for the term *"Oslo"* using the "instance of"-property, as seen in Figure 3.1.

3.1.2 Constructing the Ontology

We implement the ontology using the programming language Python[8]. We are aware of other available, traditional ontology and semantic web technologies such as *Web Ontology Language* (OWL)[9] and *Resource Description Framework* (RDF)[10]. However, we choose not to utilize these here, as we assume available data structures from Python are sufficient for the purpose and extent of this thesis.

The ontology is constructed querying Wikidata as described above, in Subsection 3.1.1.1. As PII spans in text documents typically refer to a human in some way (education, date of birth or death, workplace, languages spoken, etc.), we base the initial version of the ontology on all entities in Wikidata related to Wikidata's *"human"*-entity id (Q5). We extract these through a filtering of the dump file[11], which results in a smaller and more manageable knowledge base with less noise.

When all human-related terms are extracted we construct generalization lists for each entity, and add the results to the ontology. We utilize lists and dictionaries in Python to format the ontology as a JSON-file. The resulting

[7] https://www.npmjs.com/package/wikidata-taxonomy
[8] https://www.python.org
[9] https://www.w3.org/OWL/
[10] https://www.w3.org/RDF/
[11] The dump file was downloaded from https://www.wikidata.org on Sept. 13, 2022.

format is shown in the example in Listing 3.1. The ontology is built up as a dictionary, where each key is a term, for easier look-up of entities to generalize. The value of the key is another dictionary with `id` and `properties` as keys. The `id`-key holds the Wikidata identifier of the term, while `properties` contain a dictionary with generalization lists and the Wikidata properties that were used to extract them. For instance, the "instance of" (P31) property is used to extract the generalization lists of *"atheism"* in the example in Listing 3.1. The keywords `first` and `longest` indicate whether the generalization list is the first encountered when traversing the query results, or the longest found (if two lists are equally long, the last encountered is kept). This division allows for further use of or analysis on the options later, if necessary.

```
"atheism": {
    "id": 'Q7066',
    "properties": {
        "P31": {
            "first": [
                "world view",
                "concept"
            ],
            "longest": [
                "philosophical movement",
                "type of world view"
            ]
        },
        "P279": {
            "first": [
                "irreligion",
                "social structure",
                "structure"
            ],
            "longest": [
                "irreligion",
                "secularism",
                "world view",
                "point of view",
                "notion",
                "belief",
                "mental state",
                "condition",
                "state",
                "phenomenon"
            ]
        }
    }
}
```

Listing 3.1: JSON-excerpt of entity from ontology

3.2 Modifying the ontology

To assess the performance and coverage of the ontology, we investigate the generalization options produced by the first version of the ontology, on shorter example texts from the annotated dataset of Wikipedia biographies described in Section 4.1.1. Through manual inspection of these results, we identify several areas to improve. In particular we seek to enhance the content and coverage

of the ontology, and removing non-usable generalizations. We also note how certain semantic types can be better generalized using heuristics.

3.2.1 Expanding the Ontology

First of all, we observe that large parts of some essential categories are consistently missing from the ontology, e.g. nationalities and countries. By querying Wikidata directly with the missed terms, we notice that we can generate suggestions for the majority of these entities, with Wikidata. This indicates the existing potential of a more complete ontology with larger coverage. We therefore expand the ontology to include generalizations for all Wikidata entities that are instances of *human population* (Q33829), *ethnic group* (Q41710) and *nationality* (Q231002).

Instances of *country* (Q6256) are also added to the ontology. For this group of entities, we also add the aliases from Wikidata to allow for various spelling of country names. This means that for instance both *"the United States of America"* and *"the U.S."* are entries in the ontology with the same generalization lists. In addition, we remove some generalization suggestions for countries, as we deem them too extreme or irrelevant. In particular, we remove all generalizations more general than *"Earth"*, such as *"Earth-Moon System", "Milky Way"* and *"Virgo Supercluster"*. To compensate for this deletion, we add the terms *"country"* and *"country in <first_ generalization>"* (e.g. *"country in North America"*, where *"North America"* is the original first level of generalization), to the generalization list of every term denoting a country.

To improve the quality of the ontology, we also remove generalizations that are not considered probable or useful replacements. They hold little semantic information, and are often used to denote highly general entities in Wikidata. Examples of such removed generalizations are *"Wikidata metaclass", "first-order class", "spatio-temporal entity", "continuant"* and *"entity whose item has the given name property"*.

3.2.2 Rule-Based Generalizations

Entities of some semantic types (see Section 4.1.1), such as PERSON, are challenging to generalize without introducing a high privacy risk, as the generalizations are likely to include a considerable amount of identifying information. For words of other semantic categories, like DATETIME and QUANTITY, the generalization task is difficult because terms of these types do not necessarily have a corresponding entity in Wikidata. We therefore develop heuristics for rule-based generation of generalizations for PIIs having the semantic type PERSON, DATETIME or QUANTITY. These are described in the following, and examples of each of them are provided in Table 3.2. Entities of all other semantic types make use of the generalization lists provided in the ontology.

PERSON

To avoid all PERSON-terms being replaced by the same term, "PERSON", we implement some simple heuristics to differentiate between the personal references. Each entity is generalized to "PERSON #", where # is a number. Entities with the same personal reference in one document, should have the same assigned

Entity type	Original term	Example replacements
PERSON	Ada Lovelace	PERSON 1
DATETIME	18 July 1980	date in the 1980s, 1980
QUANTITY	13 seconds	X seconds

Table 3.2: Examples of rule-based generalizations for the semantic types PERSON, DATETIME and QUANTITY.

number throughout the text. To ensure such co-reference, we assume that entities that are either identical or sharing the same last name, are co-referent. As a consequence they will receive the same number. For instance, the PII *"Ada Lovelace"* and *"Mrs. Lovelace"* may both receive "PERSON 1" as replacement suggestion, but another PERSON-entity in the same document may be generalized to "PERSON 2". For the annotated dataset of Wikipedia biographies, we also utilize the annotated `related_mentions` item to ensure correct PERSON-numbering.

DATETIME

We differentiate between the format of various DATETIME-terms. If the PII is a year, we replace it with "date in the <decade>", where <decade> is the decade of the year. The same generalization is provided if the span is a longer date, e.g. including day and month, but for these terms, the identified year is suggested as replacement in addition.

QUANTITY

Text spans denoting a quantity are generalized to "X <unit>", where <unit> is the unit of measurement. If no unit is identified, only "X" is suggested, to indicate that the entity is a quantity.

3.3 Final Ontology

The included heuristics (Section 3.2.2) and the final, modified ontology form a system for generating replacement suggestions for PIIs in text. We do, however, emphasize that even though the ontology is enhanced, there are still remaining parts subject to improvement. We elaborate on this in Section 6.3.

Figure 3.2 provides an overview of our approach to generating suitable generalizations using the Wikidata-derived ontology and heuristics. This is further described in the following sections. First, we briefly describe the use of the system, as presented in Figure 3.2 (Olstad et al., 2023). Secondly, we elaborate on the details of this generalization process.

3.3.1 Using the System

As shown in Figure 3.2, the system takes as input a PII span, and outputs one or more generalization suggestions. We assume here that the input span is a direct or quasi identifier, that should be generalized. The process of obtaining

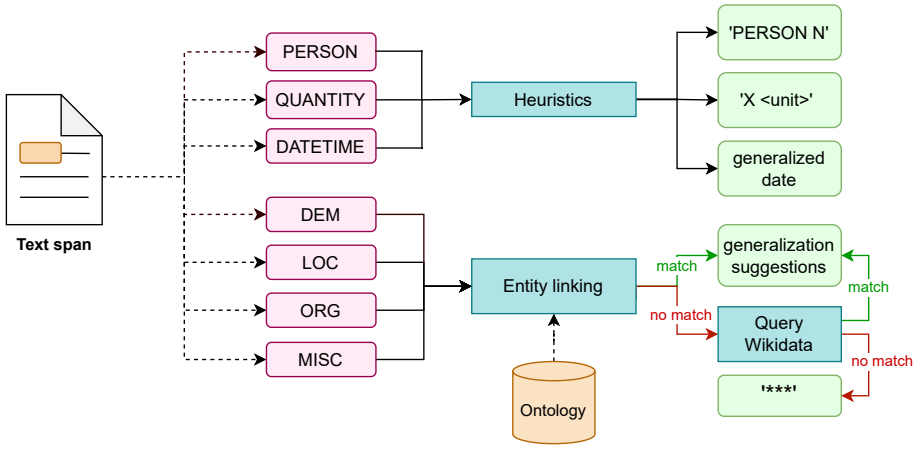

Figure 3.2: Generation of replacement options for text spans. Depending on the entity type, the replacements are produced using either heuristics or the Wikidata-derived ontology.

the replacements depends on the entity type (see Section 4.1.1) of the term to be generalized:

- Replacements for entities of type PERSON, QUANTITY and DATETIME are generated using the heuristics described in Section 3.2.2.
- LOC, ORG, DEM and MISC entities are replaced with the suggestions in the ontology. The PII-span is linked to terms in the ontology using the techniques described in Section 3.3.2. If there are no suggestions in the ontology, the system queries Wikidata directly. If there are no returned generalizations from this query, "***" is returned as only replacement option.
- Entities of the CODE type cannot be generalized due to their informativeness. They are therefore always replaced by "***".

In the following we describe the details of the system, i.e. how we traverse the ontology and link PII spans to entities in it.

3.3.2 Details of the Generalization System

To generalize entities of the semantic type LOC, ORG, DEM and MISC, we search through the ontology to find an equal or similar term with a generalization list. This entity linking is achieved through three different approaches, depending on the results from the previous matching:

Exact Match First, we search for an exact string match in the ontology.

Contained_in Match If no match is found by exact matching, we perform a *contained_in* search. For instance, we may consider *"Brussels"* and *"Brussels city"* a match, since the former is contained in the latter.

Property	# of gen.	# of times used	avg # of gen.
P31	992,058	712,769	1.4
P279	887,267	106,761	8.3
P8225	8	8	1.0
P361	3,328	1,011	3.3
Total	1,882,661	820,549	3.5

Table 3.3: Statistics on the properties used in the ontology.

Approximate String Match Lastly, if none of the above steps return a match, we employ approximate string matching to link entities. We consider it a match between two terms, if the character-level edit distance is below a strict threshold, here 15%.

Once the PII span is linked to an entity in the ontology, we retrieve the relevant generalization options. If there is more than one replacement list available, we select the first one. This means that the list produced from the P31 property is selected before the list produced by the P279 property. Lastly, P8225 or P361 is selected, in that order. All lists will include the "***"-masking as a last option.

3.3.3 Statistics on the Ontology

To assess the coverage of the final Wikidata-derived ontology, we provide in this subsection statistics on the covered concepts and the properties used for their generalization.

The total number of extracted concepts is 753,955, which means that our ontology covers more than 750,000 possible identifiers. 66,594 of these have generalization options provided by more than one property, e.g. the string *"Mayor of Saint-Étienne-à-Arnes"* having proposed replacements based on both the P31 (*"position"*) and P279 (*"mayor of a place in France"*) membership properties.

In Table 3.3 we report various statistics regarding the properties used to generalize the terms in the ontology. *# of gen.* refers to the total number of suggested replacements per property, as acquired while traversing the expanded ontology. *# of times used* is the number of times the property was used to retrieve replacement suggestions for the items in the ontology. Finally, *avg # of gen.* is the average length of the list of hierarchical replacement options per property.

We observe in the *# of times used* column that P31 and P279 are the most used properties for replacement generation in our ontology. Furthermore, we see that these two properties also produce the largest amount of generalizations, in total. We do, however, note that though the P31 property produces many generalizations in the ontology as a whole, the average number of generalizations per term (1.4) is smaller than both that of P279 (8.3) and P361 (3.3). This indicates that even though the P31 property frequently suggests replacements for the concepts, it does not necessarily propose a large number of generalizations per concept, leading to fewer replacement options to choose between.

Chapter 4

WikiReplace: Generation of Dataset

4.1 Data

Many of the available datasets for text anonymization are in the field of clinical NLP, aiming to detect PHI spans, as noted by Pilán et al. (2022) and Meystre et al. (2010). There are few publicly available, large datasets outside the medical domain. For example, one alternative is to use datasets of personal emails (e.g. the Enron email dataset[1]) to evaluate the text anonymization task (Pilán et al., 2022).

More recently, in 2022, two annotated datasets for text anonymization have been published. First, the *Text Anonymization Benchmark* (TAB) (Pilán et al., 2022), which consists of manually annotated court cases from the European Court of Human Rights (ECHR), for the purpose of text anonymization[2]. Following this annotation effort, a collection of annotated Wikipedia summaries of biographies Papadopoulou, Lison, et al. (2022) was released.[3]

In this thesis, we base our work on the latter (the annotated collection of Wikipedia summaries) and some of the work performed in (Pilán et al., 2022) (the TAB dataset). The data are further described in Subsection 4.1.1 below.

4.1.1 Annotated Wikipedia Dataset

Papadopoulou, Lison, et al. (2022) released a freely available, manually annotated dataset for text anonymization, consisting of 553 Wikipedia summaries. The annotators in this work were given two tasks:

1. detect text spans of personal information and
2. decide whether these terms should be masked or not, in order to protect the identity of the individual

[1] https://www.cs.cmu.edu/~enron/

[2] https://github.com/NorskRegnesentral/text-anonymization-benchmark/blob/master/guidelines.md

[3] https://github.com/anthipapa/textanonymization/blob/main/annotation_guidelines.pdf

Entity type	Examples
CODE	SK 4631, NOR1234, 8778/323
DATETIME	15/05/23, ten years, the following day
DEM	Norwegian, veterinarian, MSc in Medicine
LOC	Belgium, Brussels, the Eiffel Tower
ORG	Red Cross, University of Oslo, Turkish Government
PERSON	Ada Lovelace, Rihanna, John Smith
QUANTITY	$10 million, 20 km/h, 20kg
MISC	action thrillers, painting, helicopter crash

Table 4.1: Examples of PII categories

In addition, they also assigned, among others, a semantic category (called *entity type* in the dataset) to each text span, following the definitions given for the TAB corpus (Pilán et al., 2022). In our work, we consider these same eight categories for PII spans, as detailed in Pilán et al. (2022) and Olstad et al. (2023):

CODE Identifying numbers and codes

DATETIME Specific date, time or duration description

DEM Demographic attributes of an individual, such as nationality, profession or education

LOC Any named locations, such as countries, cities and named infrastructure

ORG Names of organizations and institutions

PERSON Names of individuals

QUANTITY Values denoting a quantity, such as monetary values, speed, or weight or number of items

MISC Information that do not belong to any of the other categories

Table 4.1 provides some examples of each of the categories above.

Furthermore, for each entity, the dataset has information on the type of identifier (QUASI or DIRECT), as well as the position of the text span in the document. Each annotated entity also has an `id` and `entity_mention_id`. The latter is used in cases where the reference of the PII span in the real world is mentioned more than once in the document. The referential entities are listed in the `related_mentions` item for each relevant entity. For further details on the structure of this dataset, we refer to Papadopoulou, Lison, et al. (2022).

For instance, in the excerpt below, the three terms in bold (*"World Wrestling Federation", "WWF"* and *"WWF"*) are marked as QUASI-identifiers. As they refer to the same entity in the real world, they are linked together with the `related_mentions` item.

> He also worked for the North American–based promotions the **World Wrestling Federation (WWF)** and Total Nonstop Action Wrestling (TNA) due to talent exchange programs between AAA and **WWF**.

This is a property that is used by the rule-based generalization for spans of the semantic type PERSON, as described in Section 3.2.2.

In the following sections, we elaborate on how we further enhance this dataset by adding generalization annotations for each text span.

In Section 4.1.1 we presented an annotated dataset of Wikipedia biographies (Papadopoulou, Lison, et al., 2022) - a freely available collection of Wikipedia summaries, manually annotated for the purpose of anonymization. In this section we build on this dataset to produce a human-labelled dataset on replacement choices for text sanitization.

The main motivation for creating this generalization dataset is to allow for automation of the sanitization process. As we will see in Chapter 5, we can utilize such a dataset for modeling the selection of replacement with machine learning. A further description on how we do this is provided in the mentioned chapter.

In the following sections we present our approach to creating the first ever (to the best of our knowledge) human-labelled dataset on replacement choices for text sanitization. We describe the preprocessing of data in Section 4.2 and the annotation process, including the annotation tool and guidelines, in 4.3. Finally, in Section 4.5 we present and analyze the final dataset.

4.2 Preparing the Data

We base our generalization dataset on the available annotations in the dataset of Papadopoulou, Lison, et al. (2022), and assume in this work, that these annotations are correct. In particular we consider the `entity_type` and `span_text` of each entity. Entities having the `entity_type` NO_MASK are not considered to contribute to the privacy risk in such an extent that they should be masked. Entities of this type is thus ignored, and only entities of type QUASI and DIRECT are considered subjects to generalization. However, we do note that some of the NO_MASK-entities may contain *sensitive* information, e.g. religious or political beliefs. Nevertheless, as they are not considered to introduce any privacy risk by the annotators in the previous annotation effort, meaning they will not disclose the identity of an individual, we assume they do not need to be generalized.

For each entity in the dataset, we use the replacement system described in Section 3.3 to extract generalization suggestions for the `span_text`. These generalization lists are stored in the `generalization` item of the entity, which are later displayed as available replacement options in the annotation tool (see Section 4.3.2).

For example, for the text span *"geologist"* apart from existing information like for example it being a QUASI identifier and having a DEM semantic type, we now provide a hierarchical list of suggested generalizations for it, namely *"earth scientist", "scientist", "erudite person", "person"*, and "***".

4.3 The Annotation Process

The annotation effort in this work focuses on selecting the most suitable replacement among suggested replacements. For this task, we recruit 9 annotators with various backgrounds to ensure a representative variation of the understanding of the annotation task. Less than half of the annotators are students with

knowledge in the field of NLP. The remaining have a higher education in other academic fields, among others law, computer science, pharmacy and geology. The age of the annotators ranges from early twenties to late fifties, and both men and women participate in the annotation task.

Each of the annotators are given 81 documents to annotate. 22 of these are to be multi-annotated by all annotators, while the remaining 55 documents are selected at random. In total, all 553 documents in the annotated dataset are further annotated for the purpose of generalization.

In the following, we present the annotation guidelines and describe the annotation tool developed for the purpose of this annotation process.

4.3.1 Annotation Guidelines

In short, the annotators are asked to select exactly one replacement per marked text span in each assigned document. For instance, they may be presented with the following generalization options for the term *"geologist"*, which as previously mentioned are:

geologist → [earth scientist] → [scientist] → [erudite person] → [person] → ***

If the annotator selects the second option, "[scientist]", the text before and after the annotation process may look like the following:

Original text
He is an American **geologist**.

Generalized text
He is an American **[scientist]**.

However, to ensure a mutual understanding of the annotation task among the annotators, we provide them with detailed guidelines. These include a description of how the task should be understood, clarifications of edge-cases and a guide on how to use the annotation tool. In short, there are three main steps in the annotation process:

- First, the annotators read through the entire text.
- Following that, they consider each marked text span and select its most appropriate replacement according to both the criteria of privacy and that of utility.
- Finally the annotators read through the text once more, to ensure that both criteria are fulfilled.

The guidelines as a whole, are included in Appendix A.

4.3.2 The Annotation Tool

To facilitate the annotation process for the annotators, we develop a web page to be used as the annotation tool for this specific effort. This flexible solution allows for annotation from anywhere with internet access, without the need of any installation.

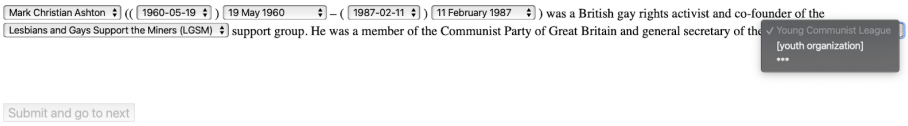

Figure 4.1: Example of document displayed in the annotation tool. The annotator selects a replacement option from the drop-down menu of each marked text span, as seen on the right.

The web page is created using the Flask framework[4]. The graphical user interface of the tool is shown in Figure 4.1. Each pre-marked PII span has a drop-down menu with suggested replacements. The annotators select the option they consider to be most suitable, both in regards to data utility and privacy. When a generalization has been selected for every span in the document, the annotators submit the annotations and continue to the next document. They are then redirected to a new page with a new document.

Each annotator is assigned a personal ID and receives a designated URL to use for the annotation. Once all files are submitted, we download the data from the server to a local computer for further processing to create the dataset. We describe this post-processing in the following.

4.4 Creating the Dataset: WikiReplace

The post-processing of the annotated data mainly consists of structuring it to a suitable format. We follow the format in the annotated dataset of Wikipedia biographies from Papadopoulou, Lison, et al. (2022), but add a `replacement`-item containing the following:

> `generalizations` : contains all proposed generalizations, i.e. the list(s) from which the annotators selected the most suitable options.
>
> `generalization_selection` : holds the selections made by each annotator. If one of the suggestions in `generalizations` was not selected, it is removed from this item.

In addition we add the item `generalized_text`, holding the generalized version of the text from the item `text`. The document is generalized according to the selections of the annotators. In the case of different selections in multi-annotated documents, the option selected by the majority of annotators replaces the corresponding PII span in the document.

Consider for instance the entity *"geologist"*. This quasi identifier has four suggested replacements, all retrieved from the P279 property (subclass of), in addition to the default "***"-masking. The annotator decided that the second level of generalization is most appropriate for this PII in the following context, taking into account all other PII in the text:

[4]https://flask.palletsprojects.com/en/2.3.x/

Entity type	Level 1		Level 2		Level > 2		***	
DATETIME	1025	(42%)	1032	(43%)	360	(15%)	764	(32%)
DEM	265	(37%)	202	(29%)	242	(34%)	318	(45%)
LOC	356	(34%)	419	(40%)	263	(25%)	524	(50%)
MISC	272	(20%)	622	(45%)	481	(35%)	964	(70%)
ORG	652	(35%)	773	(42%)	430	(23%)	1066	(57%)
PERSON	2478	(97%)	85	(3%)	0	(0%)	18	(0.7%)
QUANTITY	381	(99%)	5	(1%)	0	(0%)	5	(1.3%)
Total	5429		3138		1776		3726	

Table 4.2: Distribution of levels of generalization per semantic type.

> Donald Ross Prothero (February 21, 1954) is an American **geologist**, paleontologist, and author who specializes in mammalian paleontology and magnetostratigraphy[...]

The selection stored in the `generalization_selection` for this entity is thus *"scientist"*, which is also used as replacement in the generalized version of the text stored in `generalized_text`.

The final dataset, named *WikiReplace*, is made freely available on GitHub[5], and presented in the work of Olstad et al. (2023). It is divided into a train and test set, consisting of 453 and 100 documents respectively.

This split is selected to ensure enough documents for testing, so that the testing results of models are representative. All multi-annotated documents are included in the test set, as these include various understandings of the correct annotation. Consequently, they are useful during evaluation, since they allow us to compare the model predictions against many different, equally correct solutions.

4.5 Results

To assess the quality of the resulting annotations, we gather statistics on the various replacement selections made by the annotators. In this section we present these numbers and provide an analysis on the resulting dataset, including the agreement between the annotators.

One of the main motivations to utilize generalization as a text sanitization technique is to provide more meaningful replacements than e.g. deletion of the PII span. In this dataset, only 36% of the replacement selections are of the default "***"-option. This means that the majority of PII spans in these documents are replaceable with more informative terms than simple deletion ("***"). As a consequence, more of the semantics of the original document is maintained in the generalized version and the data utility increases accordingly. As the annotators were also asked to consider the privacy risk of each option when selecting a replacement, we may assume that the selected options do not, or to a minimal extent, introduce a privacy risk to the document.

Furthermore, we consider the selected level of generalization, as this too is related to the resulting semantic content in the generalized text. By "generalization level", we mean the position of the selected replacement in the generalization

[5]https://github.com/anthipapa/bootstrapping-anonymization/tree/main/wiki-replace

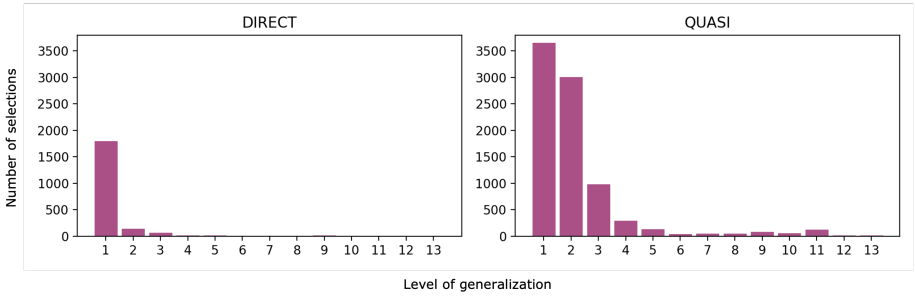

Figure 4.2: Distribution of the annotators' selection of generalization levels according to the DIRECT and QUASI identifier types

list. This corresponds to the degree of generality of the options, as explained in Section 3.1.1.1. For instance, for the PII *"geologist"* in the *"Donald Ross Prothero"* example above, *"earth scientist"* is the first level of generalization, while *"scientist"* is that of level two. The default "***"-masking is the last level of generalization, in this case level five.

Figure 4.2 shows the distribution of selected generalization levels in regards to the identifier type, i.e. DIRECT or QUASI identifiers. The annotators' tendency of selecting lower generalization levels, such as 1 and 2, is also reflected here. This is particularly true for DIRECT identifiers, having almost no selections of generalizations on levels higher than the first. The QUASI-identifiers, on the other hand, has a more disperse distribution of levels, though most selections are centered around the lower levels from 1 to 5.

This information is not informative enough since the DIRECT and QUASI categories include multiple possible semantic types that might affect the generalization level differently. For this reason, we also do analysis on the relation between semantic category and generalization level.

In Table 4.2 (Olstad et al., 2023) we report the number of selected generalization levels among all annotations, in regards to the entities' semantic types. We observe that more than half of the selections are of first level replacements, i.e. the most specific generalization suggestions. This indicates that the option holding the most semantic content, without being as specific as the original text span, is favorable in the majority of the cases. This, without introducing additional privacy risk, according to the annotators.

However, we do note that the semantic categories PERSON and QUANTITY have fewer replacement suggestions than others. Neither of these entity types have any replacement options of a level higher than 2, as reported in Table 4.2. Consequently, the possible selections for these categories only consist of two options: the default value "***" and one generalization. In the case of PERSON-entities, the generalization is the "PERSON #"-replacement provided by the heuristics, while the option of the QUANTITY-group is "X <unit>" (see Section 3.2.2 on heuristics).

As a result, these semantic categories achieve a higher percentage of low-level selections than the other entity types. For instance, as we see in Figure 4.3, 98% and 96% of the selections made for the QUANTITY and PERSON categories respectively, are of the most specific option (level 1). For the MISC-category, however, this percentage is much lower - only 20%. This imbalance in number

of options, should thus be noted when considering the levels of generalizations.

4.5.1 Inter Annotator Agreement

As described in Section 4.3, 22 of the documents are multi-annotated. We need to assess if this annotation effort has been consistent. To get an impression of how much the replacement selections vary between annotators, we compute the inter annotator agreement (IAA) using *Light's kappa* (L-kappa)(Conger, 1980). This metric is suitable for the IAA-computation of this annotation task, as it allows for for multiple annotators selecting an option from numerous alternatives. An L-kappa value of 1 indicates perfect agreement between annotators, while -1 suggests direct disagreements. To compute Light's kappa(Conger, 1980), we compute the mean value of all pairwise agreement scores for the annotators, calculated as the Cohen's kappa (κ) coefficient(Cohen, 1960). This metric for inter-rater reliability consider the possibility of chance agreement, and is given as the following formula:

$$\kappa = \frac{Pr(a) - Pr(e)}{1 - Pr(e)} \tag{4.1}$$

where $Pr(a)$ signifies the actual agreement between annotators, and $Pr(e)$ represents the agreement by chance (McHugh, 2012).

For this dataset, the L-kappa is 0.61, which indicates a moderate to substantial agreement, but also suggests a challenging annotation task where not all annotators agree on which replacement term is the most suitable. This is also reflected in the varying agreement, between annotator pairs, as seen in the confusion matrix in Figure 4.4. The agreement ranges from 0.46 to 0.85, where annotator 1 and 5 agree the most and annotator 1 and 2 the least.

This shows that though the agreement is substantial, disagreements are also present, in particular between certain annotators, such as 1 and 2, and 2 and 5. The disagreements vary along the annotator pairs, but some examples of the most common disagreements are the following:

DATE Many disagreements concern the generalization of dates. Specifically, there are numerous conflicting annotations regarding the "date in the <decade>"-alternative and the masking "***". Consider for instance the following example:

Original Text

These French colonists had established themselves in **1555** [...]

Generalized text

1. These French colonists had established themselves in **[date in the 1550s]** [...]
2. These French colonists had established themselves in **[***]** [...]

The first replacement is selected by four annotators, while five annotators choose the latter. There may be various reasons for the conflicting annotations, but possible reasons include the impact other selections made in the document and the subjective understanding of whether *"date in the 1550s"* may be identifying or not.

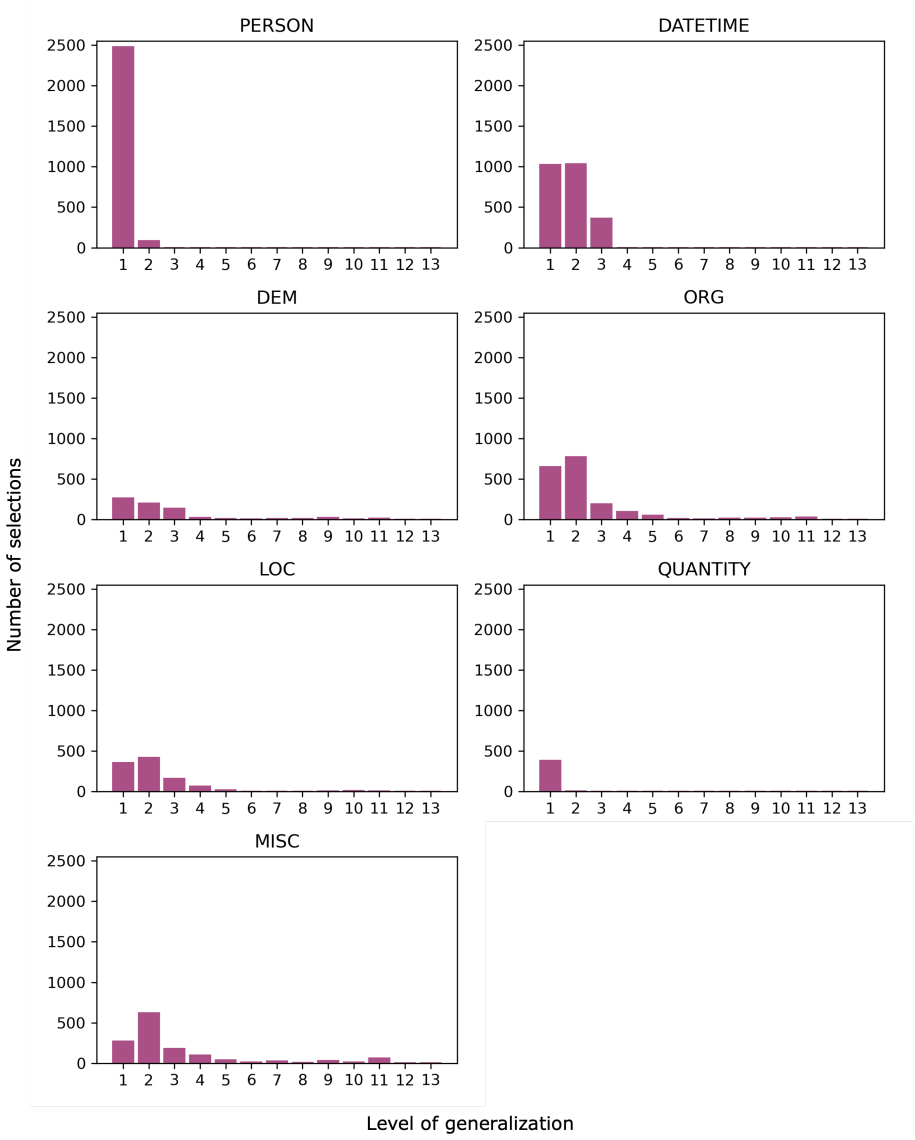

Figure 4.3: Distribution of the annotators' selection of generalization levels according to the semantic types

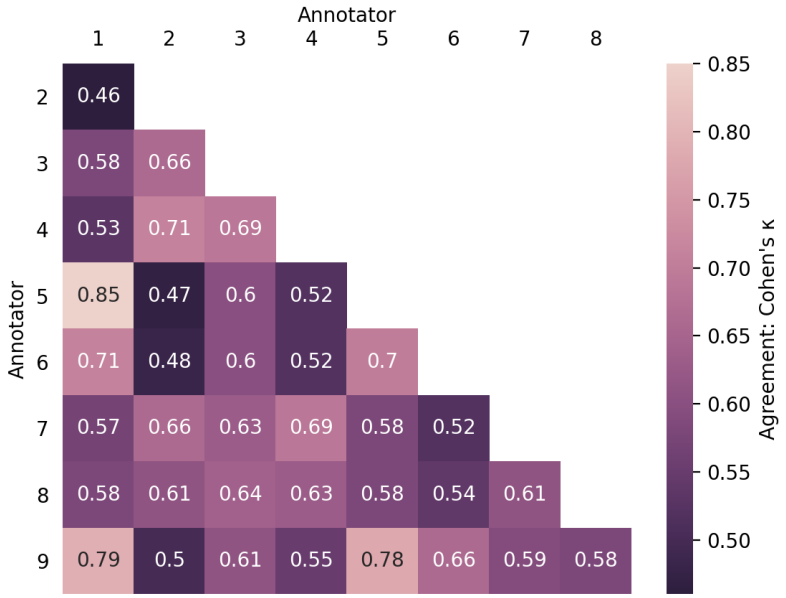

Figure 4.4: Pairwise inter-annotator agreement, computed as Cohen's kappa.

DEM There are also many examples of conflicting selections made for demographic identifiers, such as the profession *"politician"* in the following example:

Original Text

[...] a Chinese military general and **politician** of the state of Cao Wei [...]

Generalized text

1. [...] a Chinese military general and **[professional]** of the state of Cao Wei [...]
2. [...] a Chinese military general and **[worker]** of the state of Cao Wei [...]
3. [...] a Chinese military general and **[person]** of the state of Cao Wei [...]
4. [...] a Chinese military general and **[***]** of the state of Cao Wei [...]

Note that in the above example, there are other identifiers than *"politician"*, but to illustrate the annotation disagreement, we only consider the identifier *"politician"* here. This quasi identifier has four proposed generalizations, all of which are selected by at least one annotator. As for the above DATE-example, the possible reasons for the variations in level selection are many. In addition to the reasons described in the previous example, the number of alternatives may also have caused the wide spread of annotation selections. This, because there is less disagreement for entities with fewer alternatives, such as PERSON-terms(as seen in Table 4.2).

However, we emphasize that though there are some disagreements between annotators, all of their decisions are considered equally correct solutions as long

as they do not raise the privacy risk and retain as much of the text as possible, which was the goal of the annotation process. Consequently, there may be more than one correct solutions, resulting in a subjective annotation task. We elaborate on this challenge of numerous possible generalization combinations in the next section (Section 4.6).

4.6 Discussion

We reflect on specific cases that we encountered during the annotation process. Some are related to the work on which we based the annotation effort, while others are caused by the complexity of the task at hand and the produced replacement suggestions.

4.6.1 Multiple Solutions

The first challenging case was reported to us by some of our annotators who noted that they believed that specific documents they were tasked to annotate should have been previously marked for PII differently. This includes, but is not limited to, missing PII-markings. One example of such a report for a missed PII is seen in the following sentence from the original dataset:

> Lieutenant **James Victor Gascoyne** (**25 May 1892** – **1976**) was an English World War I **flying ace** credited with five aerial victories.

The terms in bold are PII the annotators of the dataset considered as necessary to mask (i.e. QUASI or DIRECT identifiers). However, some of our annotators pointed out that the phrase *"five aerial victories"* nearly is the definition of the masked term *"flying ace"*, thus containing just as much identifying information. Consequently, the annotators remarked that they would prefer to have generalizations for *"five aerial victories"* too.

Seeing how the first annotation task of the dataset has different annotation guidelines than ours, and also different annotators that took part in it, it is safe to say that such remarks show how objective the task of text sanitization is, and that it can have several equally correct solutions, as also noted and discussed in Papadopoulou, Lison, et al. (2022).

Secondly, selecting an appropriate level of generalization is a difficult task, as reflected in the moderate inter-annotator agreement (Subsection 4.5.1). The main cause of this challenge, is that there is usually more than one appropriate combination of generalizations in a document. The generalization selection for one term will necessarily affect the selection made for the next PII, as it is the total amount of information provided in a document that may lead to the re-identification of an individual. For instance, selecting a highly informative replacement for one PII in the text, may introduce the need of a more general replacement for other PII spans in the same document. Which replacements are kept specific and which are made more general may vary, allowing for various generalization versions of a document. These variations make the annotation task inherently challenging.

4.6.2 Readability and Grammaticality

Another issue related to the generalization selection task is that the readability and grammatical correctness of the text is not always preserved through generalization. This is caused by generalization suggestions that, in isolation, are appropriate replacements, but not in the context of other words. The incompatibility with surrounding tokens is typically rooted in grammatical issues. For instance, consider the following generalized text:

Original text
Reniers is a forward who was born in **Tilburg**.

Generalized text
[PERSON 1] is a forward who was born in **[big city]**.

In this example, replacing *"Tilburg"* with *"big city"* may be appropriate when looking at the term in isolation. But, when considering the replacement in the text as a whole, we observe that the sentence become ungrammatical, as the article *"a"* is missing before the replacement. A more suitable generalization, in regards to readability, could thus be:

[PERSON 1] is a forward who was born in **[a big city]**.

4.6.3 Heuristics

Another challenge the annotators encountered was that of non-suitable or missing replacements. The reason for why certain entities either lack or have irrelevant replacements, depends on the semantic type of the relevant PII. For the entity types PERSON, QUANTITY and DATETIME, these challenges are likely caused by limitations in the heuristics, as they cannot cover every possible PII.

For instance, we see that there are cases where the unit of measurement is not found for QUANTITY-terms, or where the date is not identified for the DATETIME-terms. Examples of such cases are provided in example 1 and 2 below.

Example 1

Original text
[...]spent their summers living on a **100-acre** rural converted farm[...]

Correct generalized text
[...]spent their summers living on a **[X-acre]** rural converted farm[...]

Annotated generalized text
[...]spent their summers living on a **[X]** rural converted farm[...]

Example 2

Original text
[...]he was with SV Dynamo in Altenberg, East Germany **between 1988 and 1990**.

Correct generalized text
[...]he was with SV Dynamo in Altenberg, East Germany **[between 1980's and 1990's]**.

Annotated generalized text
[...]he was with SV Dynamo in Altenberg, East Germany **[DATE]**.

In addition, we notice some errors in the co-referencing of entities in the PERSON-group, which are caused by the simplifying assumption that all entities with the same last name in a document refer to the same person (see Subsection 3.2.2). An example of such a case is shown in the third example below.

Example 3

Original text
Théodolinde de Beauharnais, [...] She was a granddaughter of **Joséphine de Beauharnais**.

Correct generalized text
[PERSON 1], [...] She was a granddaughter of **[PERSON 2]**

Annotated generalized text
[PERSON 1], [...] She was a granddaughter of **[PERSON 1]**.

The entities of other semantic categories are generalized using the ontology, and consequently have other challenges. Most of these are caused by the limitations in the ontology's coverage. However, we do note a particular challenge for certain identifiers of the MISC-type. Some of these entities are, for example, direct quotes of individuals and thus usually highly identifying. But, as they consist of numerous words, there are no ways of simply replacing them by a look-up in the ontology. This results in a high number of "***"-selections for such identifiers, as seen in the example below:

Original text
Afterwards, Haugen remarked: **"They must have been very tough taxi drivers."**.

Generalized text
Afterwards, Haugen remarked: **[***]**.

One solution to address this challenge may be to utilize a form of *automatic text summarization* to automatically rephrase and summarize the relevant text, e.g. by techniques presented in El-Kassas et al. (2021).

4.6.4 Ambiguity

Lastly, a case pointed out by several annotators was the issue of ambiguous words. Tokens having various meanings depending on the context, is a well-known challenge in NLP. In our case, this problem was particularly present for languages and nationalities. Consider for instance the term *"Norwegian"*. This word may be used for both the Norwegian language and to indicate the Norwegian nationality, e.g. of an individual. The context in which *"Norwegian"* occurs will thus necessarily affect which generalization is the most suitable, as seen in the example generalizations in Table 4.3.

We observe here that the most appropriate replacement may be e.g. *"Scandinavian"* or *"Germanic language"*, depending on whether the prepositioned verb is *"to be"* or *"to speak"*. A challenge like this could be solved by applying disambiguation techniques, e.g. by considering the semantic category of the term when generalizing.

Original	**Generalized**	**Inappropriate**
She is **Norwegian**	She is **Scandinavian**	*She is **language**
She speaks **Norwegian**	She speaks a **language**	?She speaks **Scandinavian**

Table 4.3: Examples of generalizations for the ambiguous word *"Norwegian"*.

Chapter 5

Selection of Replacement Options

A central objective of this thesis is to provide an approach for automatically sanitizing textual data using generalization techniques. As we saw in Chapter 3, we have developed a system for generating replacement suggestions. The next step in the generalization process, is to select the most appropriate option, amongst the ones provided, to mimic the annotators' task for the WikiReplace dataset (see Section 4.3). For this selection step, we utilize the resulting dataset and machine learning models.

One of the main challenges of developing machine learning models for automatic generalization of text documents, has been the lack of available training data. However, in the previous chapter (Chapter 4), we proposed a dataset, WikiReplace, aiming to resolve these limitations. In this chapter we therefore use WikiReplace to train ML models that automatically select the best generalization from a list of suggested replacements for a PII span. This can be modeled as a multiclass or a binary problem. We propose a solution for both approaches, in order to find the one performing the best for the selection of replacement task.

In Section 5.1 we describe the data and framework we use for training and evaluating models. Then, a detailed description of both the multiclass and binary models follows in Section 5.2 and 5.3, respectively. Finally, we analyze and discuss the findings from Section 5.2 and 5.3, in Section 5.4.

5.1 Data and Machine Learning Framework

The models in the following sections are trained on the WikiReplace dataset described in Chapter 4. We follow the division of train and test documents provided in the dataset, described in Section 4.4. Consequently, the models train on 453 documents, using cross-validation, while the remaining 100 documents are used for the final testing and evaluation of the models. The WikiReplace dataset contains several information pieces we will use as features. However, as

we describe in Section 5.2.1 and 5.3.1, these are of various types.

In more traditional approaches to training a ML model, achieving a good performance requires that we follow specific steps. These range from collection and pre-processing of data, to experiment and choose a suitable model and tune hyperparameters, and to selecting appropriate evaluation metrics to interpret the results. Opposite that are *automated machine learning* (AutoML) approaches whose benefits include, among others, being able to handle many different data types as well as not having to explicitly choose an algorithm or hyperparameters, since there are many different ones being trained, optimized, stacked and ensembled at run time. This means that we can build end-to-end pipelines that can take raw data as input and then train models that give good predictions, all without human input. In addition to facilitating the ML process, this often also leads to faster training and inference, with models that typically outperform more traditional ones.

AutoGluon (Erickson et al., 2020) is such a toolkit that can be used to train various ML models on different types of data in a tabular format. This type of machine learning is more suitable for this thesis, since the dataset we use to train and evaluate contains a large amount of different types of information that could be useful for learning the task. Among others, we consider number of generalization levels, input and replacement strings, and whether the replacement is "***" or not. Since the generalization task we present here also includes strings, this toolkit is chosen as it makes use of language models like ELECTRA[1] (Clark et al., 2020) or RoBERTa[2] (Devlin et al., 2019). Consequently, for the task of automatically choosing the best generalization out of multiple options, this AutoML approach is ideal, since it can produce a model that can be trained on various types of input at the same time.[3]

AutoGluon provides a number of predictors, with the most suitable one for our task being the **MultiModalPredictor**[4]. This predictor is able to select, tune and then fuse multiple models from various sources, depending on the type of data provided as input. We note here that there are many different models and combinations one could use with this framework. After experimenting with various combinations (Shi et al., 2021), the authors decided on better performing, specific models as default ones, which we also use and describe in this thesis.

For textual input, the predictor can make use of a pretrained Hugging Face [5] (HF) text transformer backbone. Shi et al. (2021) experiments with two different language models as transformer backbones for the textual input in the MultiModalPredictor. Of these, ELECTRA(Clark et al., 2020) was the best-performing one among a number of tasks for columns containing text data. Consequently, we also use this network for our text data. For categorical and numerical input, a Multi-layer Perceptron (MLP) is used. The categorical MLP,

[1] https://huggingface.co/google/electra-base-discriminator

[2] https://huggingface.co/distilroberta-base

[3] We note here that AutoGluon has specific methods and thresholds for inferring raw data types which might lead to some "errors" (e.g. a column containing strings can either be classified as *categorical data* or *text data* depending on different factors like the number of unique values etc.). We had to make some changes, like explicitly disabling the default conversion of categorical data to text data, and we also had to change the threshold so that text data were not considered categorical.

[4] https://auto.gluon.ai/stable/api/autogluon.multimodal.MultiModalPredictor.html

[5] https://huggingface.co/

which is relevant in this thesis, calculates the input dimension based on the number of categories that can be found in the column.

After each network is trained, the predictor then fuses the features from the various networks, with a fuse-late strategy, where the information from each network is aggregated near the output layer (Shi et al., 2021). This strategy is used to fuse features from different models by adapting them to specified dimensions, concatenating the output adapted features, and finally fusing them with an MLP.

In the following sections we describe how we use different inputs to frame the task as a binary problem and a multiclass one, utilizing the MultiModalPredictor described above.

5.2 Multiclass Approach

The first solution we propose for automatic selection of replacements in text sanitization, is a multiclass approach. As detailed in Section 4.5, the generalization lists produced for each detected PII in a document are sorted from least to most general. Consequently, we can refer to the selected replacement terms by their level of generalization. It is this referencing we utilize when modeling the generalization selection task as a multiclass problem, where the model is tasked with predicting the correct level of generalization for a given span in the text.

Below, we describe the input used for fitting the predictor and the models we train. Furthermore, we present the performance and results of the best model.

5.2.1 Input

The multiclass model considers the following three inputs when predicting the correct level of generalization:

1. **Semantic type**: the semantic type of the entity, given as `entity_type` in the WikiReplace dataset. There are a total of seven semantic types in the dataset.

2. **Number of generalizations**: the number of suggested replacements for the given PII, i.e. the total number of generalizations presented in the `generalizations` item in WikiReplace, as given by our replacement suggestion system.

3. **Text span**: the original text span to replace.

Table 5.1 provides examples of the input. The rightmost column, *selected_level*, holds the ground truth level selections, i.e. the values the model should predict.

An important note regarding this way of modeling the generalization selection task, is that we in a very limited extent make use of the actual text. Framing the task as a multiclass problem, we only consider the original text span and not the actual replacement term. Consequently, this approach partly lacks a more detailed language modeling aspect. In Section 5.3 we address this challenge by proposing a different model strategy.

sem.type	# gen.	text_span	selected_level
DEM	7	"drummer"	3
LOC	3	"Port Dover"	1
MISC	2	"Looney Tunes"	1

Table 5.1: Example of input used for training the multiclass model. Each row is one PII to replace, and each column is the input value. From left to right: semantic type, number of generalization options, original PII span and the correct level selection (to predict).

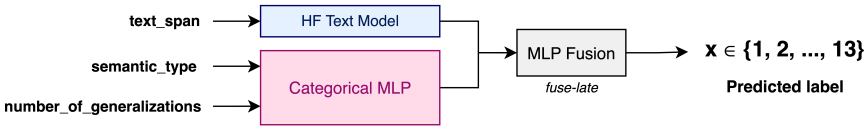

Figure 5.1: Overview of multiclass model.

5.2.2 Model

To train the multiclass model, we utilize AutoGluon's MultiModalPredictor, described in Section 5.1. For our classification task, the semantic type and the number of generalizations are inferred as categorical input, and the text span as text input. The first two are initially run through the Categorical MLP, while the text is handled by the HF_text ELECTRA model. The features are at a later stage adapted by an MLP and fused. At runtime, the multiclass model predicts the generalization level with the highest probability for a given span.

Figure 5.1 provides an overview of how the multiclass models work. Given the feature vector of a detected PII entity, the objective of the model is to correctly predict the level of generalization among all possible generalization levels. There are in total 12 possible levels (1-13) for the WikiReplace dataset, as the option with the highest level in the dataset is of level 13. The output of the model is the predicted level for each entity.

The MultiModalPredictor runs for 10 epochs with early stopping implemented with a patience of 10, while monitoring validation accuracy and a batch size of 128. The learning rate of the model is 0.0001 and Adamw (Loshchilov & Hutter, 2019) is used as an optimizer. The categorical MLP has a hidden size of 64, uses a leaky ReLU activation function, has 1 layer, and a dropout rate of 0.1. The HF_text model being based on ELECTRA uses the equivalent tokenizer and can process strings up to 512 tokens. Finally the fusion MLP model has hidden sizes of 128, is composed of 1 layer, uses the leaky ReLU activation function and a dropout rate of 0.1, similar to the Categorical MLP model. To accelerate the training time, we utilized a single GPU node.

5.3 Binary Approach

In the second approach we propose, we model the generalization selection task as a binary problem. The advantage of this modeling choice, over the previously mentioned one, is that we now can exploit the replacement string in the

span_replacement_pair	sem.type	level	star	label
drummer <SEP> percussionist	DEM	1	0	0
drummer <SEP> instrumentalist	DEM	2	0	0
drummer <SEP> musician	DEM	3	0	1
drummer <SEP> artist	DEM	4	0	0
drummer <SEP> creator	DEM	5	0	0
drummer <SEP> person	DEM	6	0	0
drummer <SEP> ***	DEM	7	1	0
Port Dover <SEP> town	LOC	1	0	1
Port Dover <SEP> classification [...]	LOC	2	0	0
Port Dover <SEP> ***	LOC	3	1	0
Looney Tunes <SEP> animated [...]	MISC	1	0	1
Looney Tunes <SEP> ***	MISC	2	1	0

Table 5.2: Examples of input used for training the binary models. Each row is one PII-replacement-pair, and each column is the feature value. Features from left to right: span-replacement-pair, semantic type, the level of the replacement in the pair, whether the replacement is "***", and lastly, whether the replacement in the pair was selected for the given PII (to predict).

prediction. This is done by including the input pairs of the original text span and the suggested replacements, where each part of the pair is separated by the separator `<SEP>`. Examples of such pairs are included in Table 5.2 (the *span_ replacement_ pair* column).

The objective of the model is to predict the correct PII-replacement-pair, i.e. for each pair determine whether it is to be selected or not. The true selected pair will have a label of 1, indicating that it was selected by the annotators, and any other pair is labeled with 0.

5.3.1 Input

The binary predictor utilizes much of the same input as the multiclass model, but also considers additional ones. In particular, the binary model makes use of all pairs of original text span and suggested replacements, as described briefly above. The binary predictor considers the following input:

1. **Span-replacement-pair**: the concatenation of the original text span and the candidate replacement, separated with the separator `<SEP>`.
2. **Semantic type**: the semantic type of the entity, given as `entity_type` in the WikiReplace dataset.
3. **Generalization level**: the level of generalization of the replacement in the span-replacement-pair.
4. **Star selection**: indicates whether the suggested replacement in the pair is "***" or not.

Examples of the input passed to the binary models are given in Table 5.2. The *label*-column, furthest to the right, contains the ground truth label selection,

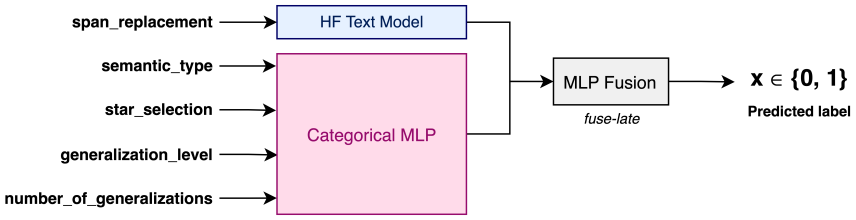

Figure 5.2: Overview of binary model.

i.e. whether the replacement in the pair was selected for the given text span in the pair.

This representation of PII for the binary model utilizes raw text to a greater extent than the multiclass model by including the replacement string in the training process alongside the original string. This is an important aspect of the task of text generalization.

5.3.2 Model

Equally to the multiclass model, we use the MultiModalPredictor for the binary model as well. Since the data are of similar types, the resulting model is also very similar to that described in Section 5.2, with all of the input being handled by a Categorical MLP, apart from the PII-replacement-pair which is handled by the LM. The features of the different models are then again fused by an MLP. At inference time, this model will give us probabilities for each possible PII-replacement pair, and we can then select the replacement with the highest probability.

An overview of the binary model is found in Figure 5.2.

The models and their hyperparameters are identical to those described in Section 5.2.2 which allows for a fair comparison between the two approaches.

5.4 Results and Discussion

In this section we present and analyze the performance and the results of the final models, both the multiclass and binary one. Following this, we further discuss and analyze the results in Subsection 5.4.2.

5.4.1 Results

We evaluate the performance of the final models using the test set of WikiReplace. In the case of multi-annotated documents, we use the majority vote of annotators to determine the gold selection. This means that we consider correct the label picked by most of the annotators for the multi-annotated documents. For instance, for the entity *"Brazil"*, one annotator selected "***" as replacement, three annotators selected *"country"* and five selected *"country in South America"*. As the majority of the annotators found the latter to be the most appropriate, we consider this to be the correct prediction.

	Accuracy	
Model	**Majority vote**	**All selections**
Baseline	51.36%	55.10%
Multiclass	70.29%	73.47%
Binary	80.05%	83.25%

Table 5.3: Obtained accuracy scores of the multiclass and binary models.

In the following we present the performance with various metrics, such as accuracy, (averaged) precision, recall and F_1-score, and the Mean Reciprocal Rank (MRR).

Accuracy The accuracy scores of the models are reported in Table 5.3. Seeing how for this task multiple different selections could be equally correct, we also compute the score considering any selection made by an annotator to be correct, not only the majority vote, as a less strict approach to evaluation.

The accuracy score is calculated directly for the multiclass model. For the binary model, we compute the accuracy based on the predicted probability of each span-replacement-pair being 1. This means that for each text span, we rank the replacements according to their probability. If the highest ranked replacement matches the gold label, we count one correct prediction for all span-replacement pairs, otherwise we deem the prediction incorrect.

For instance, considering the *"drummer"*-example given in Table 5.2: if the model correctly predicts *"musician"* as the replacement for *"drummer"*, we count this as one correct prediction, and not seven (one for each *"drummer"*-replacement-pair). This is done to ensure that the reported accuracy scores of both models are comparable.

As a sanity check of the results, we also consider the performance of a baseline "dummy" model. This model consistently predicts the overall most selected level in the training data: the first level of generalization. The baseline model achieves an accuracy of 51.36% when considering the majority vote, and a slightly higher score of 55.10% if all selections are included.

Compared to the most-frequently selected replacement choice of the human annotators, the multiclass model obtained an accuracy of 70.29%, while the binary model performed even better, with an accuracy score of 80.05%.

When considering any selected level by the annotators as a correct solution, the accuracy scores increase slightly for both models. For the multiclass it increases by 3.18%, while for the binary model this percentage is 3.2%.

Recall, Precision and F_1-score To better understand the general performance of the models on the test set, we compute the precision, recall and F_1-score for both models. Those are metrics commonly used in NLP. However, we do remark that this being a ranking task rather than an information extraction task, these metrics are less suitable, even though we frame the task as a potential binary classification problem too.

For the multiclass model we report the scores as micro-, macro- and weighted averaged in Table 5.4. Table 5.5 reports the performance of the binary model. In both tables, we also report the weighted average scores of the baseline model.

Averaging	Precision	Recall	F_1-score
Baseline	27.02%	51.98%	35.56%
micro	70.29%	70.29%	70.29%
macro	30.00%	27.67%	28.08%
weighted	67.80%	70.29%	68.74%

Table 5.4: Averaged performance scores of the multiclass model. We report the Precision, Recall and F_1-score.

	Precision	Recall	F_1-score
Baseline	27.02%	51.98%	35.56%
Score	77.56%	74.66%	76.08%

Table 5.5: Performance of the binary model. We report the Precision, Recall and F_1-score.

We consider this averaging as it accounts for label imbalance, which we will necessarily have for a baseline model consistently predicting the same label.

Mean Reciprocal Rank (MRR) Mean Reciprocal Rank (MRR) (Voorhees & Tice, 2000) is a metric evaluating the performance of a ranking system, in regards to how well it prioritizes the various options. It is computed as the mean of the Reciprocal Rank (RR) of all documents, where the score ranges from 0 to 1, with 1 indicating that the relevant document is ranked first. This score is halved to $\frac{1}{2}$ if the most relevant document is ranked second, $\frac{1}{3}$ if ranked third etc. (Craswell, 2009). It is used to evaluate search systems, question-answering tasks, and recommendation systems, among others.

The score is calculated as:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_1}$$

In our case, if the correct level of generalization is ranked first, the RR is 1, but if the ranking is lower, the RR is $1/rank$, where $rank$ is the ranking of the correct generalization. An MRR score close to 1 is thus preferable. Both ways of framing this task, as described in Section 5.2 and 5.3, can be evaluated using this metric, as they pick an appropriate replacement out of a pool of options.

Since the multiclass model estimates the probability of each level, we can use MRR to evaluate the ranking this model predicts. Similarly, we can compute the MRR-score of the binary model by ranking the `span_replacement_pair`s for each span, and compare the results to the gold labels. Table 5.6 shows the MRR score for both models.

	Multiclass model	Binary model
MRR score	0.83	0.89

Table 5.6: Performance of the binary and the multiclass models. MRR scores close to 1 denote a model that ranks the correct level higher on the list each time.

5.4.2 Analysis and Discussion

Model performance Regarding accuracy, Table 5.3 shows that both models perform better than our baseline, meaning that they were actually able to learn the task without defaulting to predicting the most frequent outcome. When compared to one another, we see that the binary model outperforms the multiclass one by almost 10%, meaning that a large majority of all predictions the model made were correct more often than for the multiclass one. When considering any annotator's option as correct, the difference in accuracy is not large enough, which means that the models, when predicting wrong, did not pick any of the level options that the annotators chose.

The multiclass model, generally having a lower performance than the binary, manages to correctly predict the generalization level for almost three out of four entities, according to the achieved accuracy. This is better than random guessing, and better than the baseline model. Nevertheless, an accuracy of roughly 70% also suggests that more than every fourth entity is replaced wrongfully. Depending on whether the model selects more or less general replacements, these erroneous selections may introduce an unwanted privacy risk.

Regarding precision, recall and F_1-scores, as shown in Table 5.4 for the multiclass and Table 5.5 for the binary one, we also note that the latter performs much better than the former, which is also consistent with the performance in accuracy. The achieved F_1-scores (76.08% and 68.74%) indicate a moderate number of false positives and false negatives. Furthermore, both models once again outperform the baseline. However, as this task mainly focuses on correctly ranking possible replacements, rather than e.g. extracting information, F_1-score, precision and recall are not essential for evaluating this task, as previously mentioned.

Finally, regarding MRR we see a change in the difference between the performance of the models, which is much smaller compared to the metrics computed and discussed before. This gives us a more informative look into the models' predictions. The models have a difference in performance of only 0.06 points when considering the MRR (Table 5.6). That means that both models are able to rank generalization options in a manner where the correct level, as selected by the annotators, is among the top ranking levels. From the rest of the performance metrics we observe that the multiclass model predicts more incorrect replacements, however, the MRR score shows that the model is able to rank the gold level among the top selections. Accordingly, the binary model makes fewer erroneous predictions in general, but when it does, one can deduce that the difference in the rank of the level seems to be larger.

In the following example, we see that the multiclass model ranked the gold prediction in second place, with the first one being incorrect. That would negatively affect the accuracy score for example, but the RR score would be halved to $\frac{1}{2}$, thus punishing the total MRR score less compared to the accuracy.

Original text

[...]a former Nigerian **senator**[...]

True generalized text

[...]a former Nigerian **[member of parliament]**[...]

Top 3 ranked generalized text

[...]a former Nigerian **[legislator]**[...]

[...]a former Nigerian **[member of parliament]**[...]
[...]a former Nigerian **[politician]**[...]

The same can be observed for the binary model. Once more the model did not predict the correct level in the first rank ([***] instead of [policy]), but it did rank it in the second place, resulting in a better MRR score.

Original text
[...]During Ford's presidency, **foreign policy** was characterized in procedural terms[...]

True generalized text
[...]During Ford's presidency, **[policy]** was characterized in procedural terms[...]

Top 3 ranked generalized text
[...]During Ford's presidency, **[***]** was characterized in procedural terms[...]
[...]During Ford's presidency, **[policy]** was characterized in procedural terms[...]
[...]During Ford's presidency, **[public policy]** was characterized in procedural terms[...]

These examples clearly show why a score like MRR is more suitable for a task like this one, and how it is much more informative for the model's actual performance than commonly used metrics like precision or accuracy.

Confusion Matrices To better understand both models, we calculate the confusion matrices of their predictions. The confusion matrix in Figure 5.3 shows which levels the multiclass approach confuses with others. We note that it is mostly lower generalization levels that are confused with each other, e.g. level 1 and 2. Generalizations on levels close to each other are expected to have a smaller difference in specificity than those with a larger number of levels between them, and are therefore less risky "errors", in regards to identity disclosure. The confusion in the model's predictions is mostly centered around nearby-laying generalization levels. In addition, there is a larger number of selections made for the lower levels (see Section 4.5), resulting in a higher probability of selections around the levels 1, 2 and 3. However, there are also some confusion between levels that are very different, such as 2 and 11.

The confusion matrix for the binary model is shown in Figure 5.4. This reports the predictions for the span-replacement-pairs in the binary model. There are 381 false positives and 447 false negatives. The remaining are correctly predicted.

Erroneous predictions Through manual inspection of a sample of the predictions, we find that the errors the models make are at times similar, while for other cases the predictions differ. For instance, neither of the models correctly predicted the replacement for *"Fianna Fáil"* in the following example:

Original text
[...]a former Irish **Fianna Fáil** politician[...]

True generalized text
[...]a former Irish **** politician[...]

Predicted generalized text
*[...]a former Irish **party leader** politician[...]

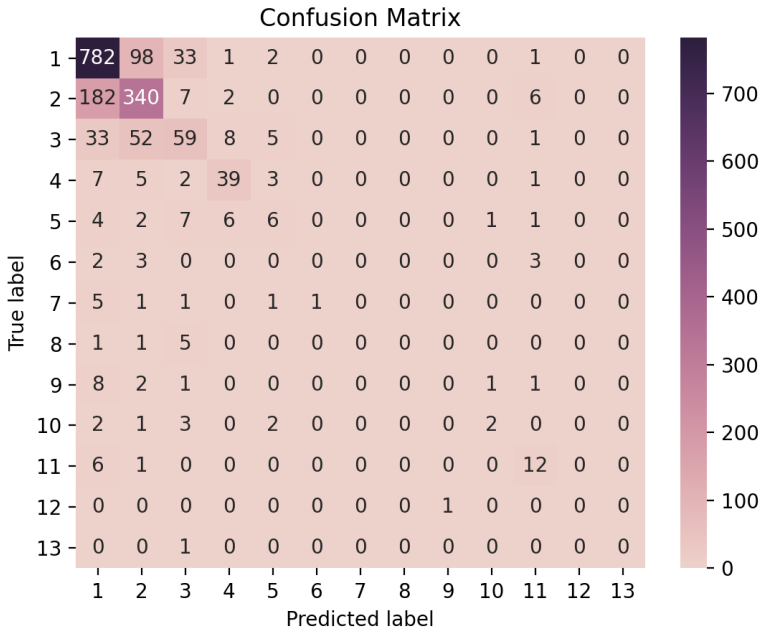

Figure 5.3: Confusion matrix of the generalization level predictions made by the multiclass model.

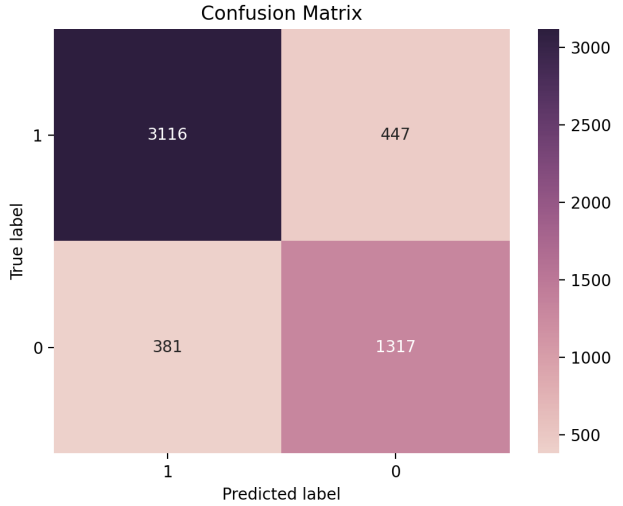

Figure 5.4: Confusion matrix of the replacement predictions made by the binary model.

Both models wrongfully predicts *"party leader"* (level 1 of 6) as replacement. This confusion may be caused by the fact that *"Fianna Fáil"* is related to the political domain, as it is a political party, but *"Fianna Fáil"* is not a politician.

There are, however, other identifiers where the models disagree on the replacement, resulting in one model making the correct selection while the other does not. Below is an example of the binary model making the correct prediction and the multiclass the wrong one:

Original Text
[...]used as a logo by **Sony Music Taiwan**.

True generalized text
[...]used as a logo by **company**.

Predicted generalized text
Binary model: [...]used as a logo by **company**.
Multiclass model: *[...]used as a logo by ***.

In this example, the binary model correctly selects *"company"* (level 1 of 3) as replacement. The multiclass model, on the other hand, generalizes too much and selects the wrong replacement: "***" (level 3 of 3). A small tendency towards stricter replacements in the multiclass model is also seen in the following example, where the multiclass model makes the correct prediction, while the binary fails to do so:

Original Text
He directed and wrote or co-wrote the films Bad Dreams, Threesome, The Craft, Dick, **Nancy Drew** [...]

True generalized text
He directed and wrote or co-wrote the films Bad Dreams, Threesome, The Craft, Dick, *** [...]

Predicted generalized text
Binary model: *He directed and wrote or co-wrote the films Bad Dreams, Threesome, The Craft, Dick, **literary character** [...]
Multiclass model: He directed and wrote or co-wrote the films Bad Dreams, Threesome, The Craft, Dick, *** [...]

As seen in the example, the binary model wrongfully predicts *"literary character"* (level 1 of 2) as replacement, while the multiclass model selects the correct alternative, which here is the stricter "***" (level 2 of 2). These differences in predictions, though erroneous, show how the framing of the modeling task has a direct impact on the final result[6].

Feature Importance An important part of using machine learning models for prediction, is to ensure that the final models are *explainable*. One way to achieve this is to be able to determine and explain how much each feature affects the models' decisions. In the following, we therefore analyze and discuss the performance of the models trained without each of the features, compared to the two original models (multiclass and binary). We do this by training the

[6] As in previous examples, we emphasize that there are more than only the highlighted PII in the above example, but in this example we only consider the identifier in bold.

Metric		Removed feature: original	semantic_type	number_of_gen.	text_span
Accuracy		70.29%	70.58%	65.25%	71.26%
Precision	*micro*	70.29%	70.58%	65.25%	71.26%
	macro	30.00%	36.11%	14.63%	23.93%
	weighted	67.80%	69.57%	59.68%	69.36%
Recall	*micro*	70.29%	70.58%	65.25%	71.26%
	macro	27.67%	37.57%	12.64%	26.80%
	weighted	70.29%	70.58%	65.24%	71.26%
F_1**-score**	*micro*	70.29%	70.58%	65.25%	71.26%
	macro	28.08%	35.50%	12.24%	23.94%
	weighted	68.74%	69.84%	60.77%	69.95%

Table 5.7: Feature importance in the multiclass model measured by averaged performance scores. A colored cell indicates a score higher than that of the original model.

models without each one of the features at a time, and report the performance. The models are trained three times, and the reported performance is the averaged scores of all three runs. In the following we refer to the previous multiclass and binary models, trained with all their respective features, as the *"original"* models.

If removing the input feature leads to a higher score, this means that the model performs better when ignoring the specified feature, i.e. the feature has a negative impact on the model's performance. The size of the difference indicates how much the feature affects the model: a higher difference is correlated to a more negative impact. On the other hand, if the removal of the relevant feature leads to a decrease in the model's performance, it indicates that the left-out feature was important for the model to make correct predictions.

Table 5.7 reports the achieved evaluation scores when removing each feature in the multiclass model. The colored cells report a score higher than that of the original model. The importance of each feature varies with the models. For the multiclass model, we observe that all scores are higher when fitting the model without the `semantic_type`. This also holds for the model without `text_span` as input, except for the macro-averaged scores. Furthermore, the model ignoring the feature `number_of_generalizations` performs worse in regards to all metrics. The decrease in accuracy of this model is reflected in Figure 5.5. This plot shows how the performance of the models without various features deviates from the accuracy of the original multiclass model.

Table 5.8 includes the achieved accuracy scores of the binary model trained without each of the five input columns. We note that the model performs better in regards to accuracy and precision when removing `star_selection`. Leaving out `generalization_level` causes an increase in the recall score of the model. Finally, removing the remaining features `semantic_type`, `span_replacement` and `number_of_generalizations`, one at a time, result in a decrease in all evaluation metrics.

Figure 5.6 visualizes how the accuracy of the binary models without certain features deviates from the originally achieved score. The removal of the feature `star_selection` leads to an increase in accuracy. We also observe a clear

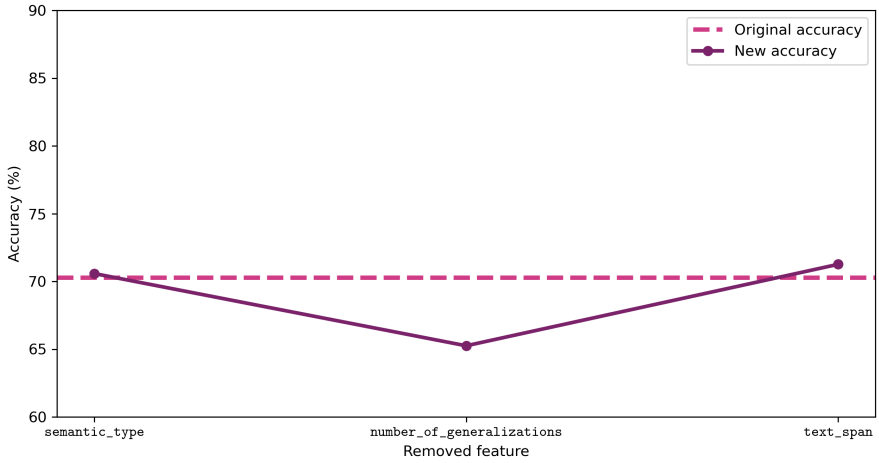

Figure 5.5: Comparison of original accuracy and the accuracy scores achieved when removing each feature in the multiclass model

Removed feature	Accuracy	Precision	Recall	F_1-score
`original`	80.05%	77.56%	74.66%	76.08%
`semantic_type`	79.19%	76.53%	73.19%	74.82%
`span_replacement`	71.41%	77.53%	64.74%	70.56%
`number_of_gen.`	79.79%	77.39%	72.96%	75.11%
`gen._level`	79.07%	75.73%	76.25%	75.99%
`star_selection`	80.39%	78.67%	73.58%	76.04%

Table 5.8: Feature importance in the binary model measured by averaged performance scores. The scores reported in colored cells are higher than the scores obtained with the original model.

drop in accuracy when leaving out the `span_replacement` feature, indicating its importance to learning the task.

In Table 5.7 and 5.8 we consider accuracy, precision, recall and F_1-score. In Figure 5.5 and 5.6 we consider the accuracy in particular, as it provides a good general overview of the models' performances and makes it easier to compare them.

In both Table 5.7 and Figure 5.5 we observe that the multiclass models without `semantic_type` and `text_span` perform better than the original model. This means that these inputs may confuse the model. `number_of_generalizations` on the other hand, is important for the model to make correct predictions. This importance is reflected in the decrease in accuracy when removing the feature: a drop by 5.04%.

The models' dependence on the `number_of_generalizations` feature is further emphasized when considering the other metrics. All scores increase when ignoring the feature `semantic_type`, and nearly all scores increase when removing `text_span`. But, the opposite happens when leaving out `number_of_generalizations`: the performance decreases by between 5% to 15%. Such a drop in

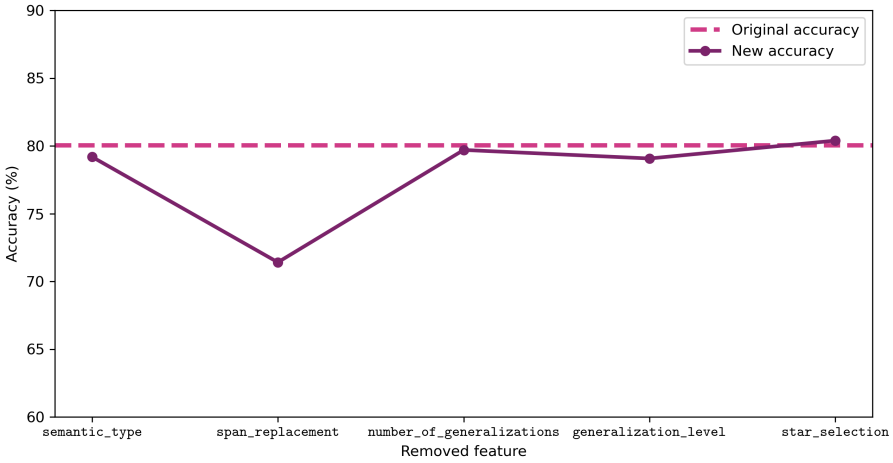

Figure 5.6: Comparison of original accuracy and the accuracy scores achieved when removing each feature in the binary model

performance underscores the importance of the `number_of_generalizations` feature for the multiclass model. Intuitively, this makes sense from a decision-making perspective: in order to correctly predict the most appropriate level, it is of great interest to know the number of possible levels to choose from.

Table 5.8 and Figure 5.6, report the performance of the binary models when removing each of the five features from the original model. We observe that removing the features `semantic_type`, `number_of_generalizations`, `generalization_level` and `star_selection` have little effect on the performance of the model. Leaving `span_replacement` out of the model, on the other hand, heavily decreases the accuracy by more than 8%. This feature is thus essential for prediction in the binary model, which is also reflected in the decrease of all the other performance scores, though to a smaller extent.

Conclusion Minimizing the privacy risk during generalization is one of the main objectives of this thesis. However, as noted, there are more than one correct solution for generalizing a document. Manual inspection shows true erroneous predictions, but also predictions of other levels that may be correct, though considered incorrect when compared to the annotators' choice. As the models are trained and evaluated on the gold data, there are cases where the models predict correct solutions that were not selected by the annotators. These predictions are, however, considered incorrect in the computation of the performance, which may yield a misrepresentative impression of their performances. The MRR score balances this out by allowing for an evaluation that takes into account the ranking of the generalization levels, instead of just the most probable one. Including other evaluation techniques, such as re-identification attacks aiming to re-identify individuals from the generalized text, may give an even stronger indication of the models' performances. This challenge in evaluating the models does, however, highlight the subjective nature of the generalization

task.

Nevertheless, considering the performances reported above, we see an indication towards the binary model predicting generalizations more aligned with the annotators' selection. As we have noted, this type of framing of the task includes a more linguistic oriented aspect (e.g. replacement span text as input during training), while we can argue that the multiclass model approach does not entail all important aspects of the task of generalization. Nevertheless, both models seem to perform almost equally well when the ranking of the levels is taken into account, as shown at the beginning of Section 5.4.2.

Chapter 6

Conclusion

6.1 Summary

Chapter 1: Introduction In Chapter 1 we introduced the objective and motivation for this thesis: utilizing generalization for text sanitization. We also defined our four research questions:

> **RQ1:** How can we create an ontology that suggests appropriate hierarchical replacements for different types of PII?
>
> **RQ2:** Subsequently, how can we use this ontology to suggest replacements and set up an annotation task to manually annotate and release a dataset of possible text replacement choices?
>
> **RQ3:** How can we use this data to choose and train a model to automatically select the best replacement for a token, without increasing the privacy risk, yet keeping a high data utility?
>
> **RQ4:** How can we evaluate such task both in regards to privacy risk, but also the resulting data utility of the text?

Chapter 2: Background In this chapter we presented the relevant terminology and previous work on sanitization and generalization. In addition, we considered the the annotated dataset of Wikipedia biographies from Papadopoulou, Lison, et al. (2022), which we used as basis for the dataset developed in this thesis.

Chapter 3: Designing the Ontology Chapter 3 described our approach to creating an ontology for text generalization. This included exploring the knowledge base (Wikidata) used to create the ontology, as well as other relevant tools.

Chapter 4: Creating a Human-Annotated Generalization Dataset In this chapter we presented the WikiReplace dataset: a human-annotated dataset for generalization, consisting of 553 documents. Chapter 4 included both a description of the structure of the dataset, and the human

annotation effort leading up to WikiReplace. We discussed the challenges related to the annotation process, and commented on the results and the inter annotator agreement. With an L-kappa score of 0.61, we saw that the task of selecting an appropriate level of generalization is a challenging, yet manageable task for human annotators.

Chapter 5: Modeling the Selection of Generalization Options In Chapter 5 we modeled the task of selecting appropriate generalizations for detected PII spans. We modeled this task in two ways: one multiclass and one binary problem. For the training of models, we utilized automatic machine learning. In particular, we used AutoGluon's MultiModalPredictor to train both models. Furthermore, we evaluated the models, and analyzed and discussed the results.

The overall best performing model was the binary model, receiving an accuracy (majority vote) of 80.05%, compared to the multiclass model's score of 70.29%. However, we noted only a small difference in the achieved MRR scores. The promising results of the models indicate that the task of automatically selecting the most appropriate generalizations given a text with detected PII, is possible.

6.2 Contributions and Limitations

In this thesis, we explored the task of generalization as part of the text sanitization task. We made several contributions. First of all, we created both an ontology and a system proposing generalizations for detected PII in text based on their semantic types. Furthermore, we published the first freely available, human annotated generalization dataset for text sanitization: WikiReplace. Utilizing this dataset, we presented two approaches to model the automatic selection of generalizations.

Initially, we presented our four research questions, all of which have been answered in this thesis. We first answered **RQ1** in Chapter 3, where we constructed an ontology utilizing the hierarchical structure of entities in Wikidata. We then answered **RQ2** through our annotation effort, presented in Chapter 4. We conclude that we can manually annotate and release a dataset of possible text replacements by combining our proposed generalization system (Chapter 3) with an annotation tool developed for the purpose of generalization selection. The annotations of the recruited annotators were then post-processed into a suitable format for the final dataset.

In Chapter 5, we answered **RQ3** by proposing two approaches to train a model for automatic selection of replacements, balancing both the privacy risk and data utility. In the same chapter we also answered **RQ4**. We evaluated the performance of the models through various evaluation metrics. Utilizing these assumes that the annotations made by the human annotators forms a gold standard, inherently evaluating both data privacy and readability of the generalized texts.

The time limit and scope of this master thesis dictated certain limitations on the work conducted. In particular, we did not experiment with other models and language models than the ones provided by AutoGluon. Furthermore, there

are other remaining issues which we present for future work in the next section.

6.3 Future Work

For future work, we have various propositions. The overall goals of these are the general enhancement of performance of the generalization system, and possible directions for the task.

First of all, we propose further enriching the ontology, as this is expected to lead to a larger coverage of generalization suggestions for identifiers and fewer "***"-replacements. Various approaches can be utilized for this. It would for instance be interesting to add Wikidata-aliases for all entries (as was done for countries, see Section 3.2). Other alternatives based on previous work include linking the ontology with a lexical database, e.g. WordNet, as proposed by McCrae and Cillessen (2021), or utilizing other resources, e.g. GeoNames (Volodina et al., 2020).

In Section 4.6, we saw that ambiguous words may pose a challenge during generalization annotation. Addressing this through various disambiguation techniques would be of high interest, as it is likely to improve the overall performance of a generalization system. One alternative is to consider the semantic category of the term when annotating for generalization.

As the annotation effort in this thesis was based on the the annotated dataset of Wikipedia biographies presented in Papadopoulou, Lison, et al. (2022), a previous detection of PII spans was assumed. To form a more comprehensive end-to-end system, combining a system for the detection of identifiers with the generalization system presented in this thesis would be of interest. Papadopoulou, Yu, et al. (2022) proposes one approach to create such a PII-detection system.

For future work, we also propose further assessment of the models' performance, e.g. by applying the generalization models to a different domain, which may yield a more correct impression of the performance. Specifically, because Wikidata and Wikipedia are similar in their content, though not in their structure. We would for instance consider applying the generalization system to the TAB corpus (Pilán et al., 2022) which is of the legal domain.

Finally, it would be interesting to experiment further with the modeling of the generalization selection task. In particular, to look at other ways to frame and model this task. For instance, it would be interesting to see if a fully BERT-based approach (similar to the work of Zhou et al. (2019), presented in Subsection 2.4.1.2) would improve the performance of the task.

Bibliography

Abián, D., Guerra, F., Martínez-Romanos, J., & Trillo-Lado, R. (2018). Wikidata and DBpedia: A Comparative Study. In J. Szymański & Y. Velegrakis (Eds.), *Semantic Keyword-Based Search on Structured Data Sources* (pp. 142–154). Springer International Publishing.

Anandan, B., Clifton, C., Jiang, W., Murugesan, M., Pastrana-Camacho, P., & Si, L. (2012). T-Plausibility: Generalizing Words to Desensitize Text [Place: Bellaterra, Catalonia, ESP Publisher: IIIA-CSIC]. *Trans. Data Privacy*, *5*(3), 505–534.

Arefyev, N., Sheludko, B., Podolskiy, A., & Panchenko, A. (2020). A Comparative Study of Lexical Substitution Approaches based on Neural Language Models [arXiv: 2006.00031]. *CoRR*, *abs/2006.00031*.

Carrell, D., Malin, B., Aberdeen, J., Bayer, S., Clark, C., Wellner, B., & Hirschman, L. (2013). Hiding in plain sight: Use of realistic surrogates to reduce exposure of protected health information in clinical text [Publisher: Oxford University Press (OUP)]. *J. Am. Med. Inform. Assoc.*, *20*(2), 342–348.

Chakaravarthy, V. T., Gupta, H., Roy, P., & Mohania, M. K. (2008). Efficient techniques for document sanitization. *Proceedings of the 17th ACM conference on Information and knowledge management*, 843–852.

Clark, K., Luong, M.-T., Le, Q. V., & Manning, C. D. (2020). Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Cohen, J. (1960). A coefficient of agreement for nominal scales [Publisher: Sage Publications Sage CA: Thousand Oaks, CA]. *Educational and psychological measurement*, *20*(1), 37–46.

Conger, A. J. (1980). Integration and generalization of kappas for multiple raters. [Publisher: American Psychological Association]. *Psychological Bulletin*, *88*(2), 322.

Craswell, N. (2009). Mean Reciprocal Rank. In L. LIU & M. T. ÖZSU (Eds.), *Encyclopedia of Database Systems* (pp. 1703–1703). Springer US.

Cumby, C. M., & Ghani, R. (2011). A Machine Learning Based System for Semi-Automatically Redacting Documents. *IAAI*.

Dalianis, H. (2019). Pseudonymisation of Swedish Electronic Patient Records Using a Rule-Based Approach. *Proceedings of the Workshop on NLP and Pseudonymisation*, 16–23.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding [Number: arXiv:1810.04805 arXiv:1810.04805 [cs]].

Domingo-Ferrer, J., Sánchez, D., & Soria-Comas, J. (2016). *Database Anonymization*. Springer International Publishing.

El-Kassas, W. S., Salama, C. R., Rafea, A. A., & Mohamed, H. K. (2021). Automatic text summarization: A comprehensive survey. *Expert Systems with Applications*, *165*, 113679.

Elliot, M., Mackey, E., O'Hara, K., & Tudor, C. (2016). *The Anonymisation Decision-Making Framework*. UKAN.

Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M., & Smola, A. (2020). AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data [arXiv:2003.06505 [cs, stat]].

GDPR. (2016). General Data Protection Regulation
European Union Regulation 2016/679.

HIPAA. (1996). Health Insurance Portability and Accountability Act.

Lison, P., Pilán, I., Sanchez, D., Batet, M., & Øvrelid, L. (2021). Anonymisation Models for Text Data: State of the art, Challenges and Future Directions. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 4188–4203.

Loshchilov, I., & Hutter, F. (2019). Decoupled Weight Decay Regularization [_eprint: 1711.05101].

McCarthy, D., & Navigli, R. (2009). The English lexical substitution task. *Language Resources and Evaluation*, *43*(2), 139–159.

McCrae, J. P., & Cillessen, D. (2021). Towards a Linking between WordNet and Wikidata. *Proceedings of the 11th Global Wordnet Conference*, 252–257.

McHugh, M. L. (2012). Interrater reliability: The kappa statistic [Publisher: Medicinska naklada]. *Biochemia Medica*, *22*(3), 276–282.

Meystre, S. M., Friedlin, F. J., South, B. R., Shen, S., & Samore, M. H. (2010). Automatic de-identification of textual documents in the electronic health record: A review of recent research. *BMC Medical Research Methodology*, *10*(1), 70.

Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the ACM*, *38*(11), 39–41.

Olstad, A. W., Papadopoulou, A., & Lison, P. (2023). Generation of Replacement Options in Text Sanitization. *The 24rd Nordic Conference on Computational Linguistics*.

Papadopoulou, A., Lison, P., Øvrelid, L., & Pilán, I. (2022). Bootstrapping text anonymization models with distant supervision. *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 4477–4487.

Papadopoulou, A., Yu, Y., Lison, P., & Øvrelid, L. (2022). Neural Text Sanitization with Explicit Measures of Privacy Risk. *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 217–229.

Party, A. 2. D. P. W. (2014). Opinion 05/2014 on Anonymisation Techniques.

Pilán, I., Lison, P., Øvrelid, L., Papadopoulou, A., Sánchez, D., & Batet, M. (2022). The Text Anonymization Benchmark (TAB): A Dedicated Corpus and Evaluation Framework for Text Anonymization [arXiv: 2202.00443]. *arXiv:2202.00443 [cs]*.

Samarati, P., & Sweeney, L. (1998). Protecting privacy when disclosing information: K-anonymity and its enforcement through generalization and suppression [Publisher: technical report, SRI International].

Sánchez, D., & Batet, M. (2016). C-sanitized: A privacy model for document redaction and sanitization: C-Sanitized: A Privacy Model for Document Redaction and Sanitization. *Journal of the Association for Information Science and Technology*, *67*(1), 148–163.

Shi, X., Mueller, J., Erickson, N., Li, M., & Smola, A. (2021). Multimodal AutoML on Structured Tables with Text Fields. *8th ICML Workshop on Automated Machine Learning (AutoML)*.

Sweeney, L. (2000). Simple Demographics Often Identify People Uniquely.

Turki, H., Shafee, T., Hadj Taieb, M. A., Ben Aouicha, M., Vrandečić, D., Das, D., & Hamdi, H. (2019). Wikidata: A large-scale collaborative ontological medical database. *Journal of Biomedical Informatics*, *99*, 103292.

Volodina, E., Ali Mohammed, Y., Derbring, S., Matsson, A., & Megyesi, B. (2020). Towards Privacy by Design in Learner Corpora Research: A Case of On-the-fly Pseudonymization of Swedish Learner Essays. *Proceedings of the 28th International Conference on Computational Linguistics*, 357–369.

Voorhees, E. M., & Tice, D. M. (2000). The TREC-8 question answering track. *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00)*.

Weitzenboeck, E. M., Lison, P., Cyndecka, M., & Langford, M. (2022). The GDPR and unstructured data: Is anonymization possible? *International Data Privacy Law*, *12*(3), 184–206.

Zhou, W., Ge, T., Xu, K., Wei, F., & Zhou, M. (2019). BERT-based Lexical Substitution. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3368–3373.

Appendix A

Annotation Guidelines

Below, we include the annotation guidelines provided to the annotators. These guidelines thoroughly describe the annotation task, including examples, and how to use the annotation tool.

Replacement Choices in Text Sanitization: Annotation Guidelines

This annotation effort is part of a larger research project that seeks to understand how to automatically remove personally identifiable information from text documents (a problem called *text sanitization*). Personally identifiable information refers to any piece of information that may directly or indirectly reveal the identity of a particular individual. Text sanitization is an important problem when dealing with sensitive documents where we need to conceal the identity of given person(s) to protect their privacy.

The result of your annotation work will be included in a new, public dataset released under an open-source license.

The Task

In this task, you are given a number of short biographies extracted from Wikipedia. To conceal the identity of the individual described in the biography, some text spans have already been marked as needing to be replaced. Each text span is shown in a drop-down menu where the values correspond to possible replacements. The original text span for which you will choose a replacement is also provided to help in the decision making process.

Your job is to select in each drop-down menu the best replacement for the text span according to the following two criteria:

1. The replacement should not disclose (directly or indirectly) the person's identity.
2. Provided that the above criteria is satisfied, the replacement should be as informative and readable as possible.

For example, in the sentence:

PERSON 1 lives and works in ***Oslo*** *...*

possible choices for ***'Oslo'*** might include *[capital of Norway]*, *[city in Norway]*, *[city]* and *'***'*. The first choice is not general enough since it is as informative as the word Oslo. The second choice is more general, followed by the third choice, and finally the default *'***'*, which is least informative (but also least risky from a privacy perspective). Person names are by default replaced by *PERSON X* (where X is an integer).

Procedure

The annotation work consists of the following steps:

- **Step 1** Read through the text once.
- **Step 2** For each marked span, look at the list of possible replacements and pick the most appropriate one. Only one replacement can be selected for each text span.
- **Step 3** When you are done with all replacements, review the text one final time. The selected replacements should not disclose the person identity, and the text should be as informative and readable and possible.

Many suggested replacements will be incorrect or irrelevant – this is normal and expected. If none of the suggested replacements are suitable for a given text span, you should choose the default '***' option.

The '***' option

In all the dropout lists of possible replacements, there will be an *'***'* option. Use this if you find that no other replacement is appropriate.

Sometimes the *'***'* is the only suitable option, since you might encounter cases where the automatic generation of suggested replacements failed to come up with good options.

Corner cases

There might be cases where a replacement looks appropriate but does not entirely fit the form of the sentence. For example, in the following sentence:

PERSON 1 was born on ***May 18, 1943*** *[...]*

The possible replacements will be *[1943]*, *[date in the 1940s]* and *'***'*. The most suitable choices in this case are *[1943]* and *[date in the 1940s]* (although it might necessitate some rephrasing to fit the current form of the sentence), not *'***'*.

Example

Below you will find a step-by-step example of the annotation steps.

Start by briefly reading the text **(Step 1)**

Then for each of the spans choose one replacement **(Step 2)**. Following is a possible set of replacements chosen.

For example, the two decades could be replaced with the *'***'* option since they provide additional information along with the rest of the personal information still left in the text (e.g. *British, gay rights activist, general secretary* etc.) that could lead to the person being identified easier, which we wish to prevent.

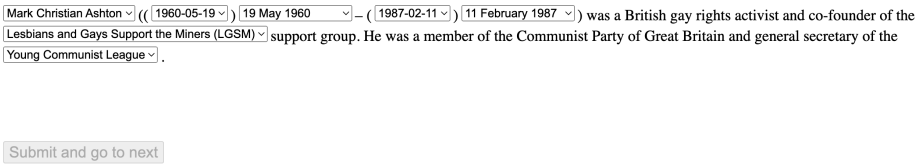

Figure 1: Step 1

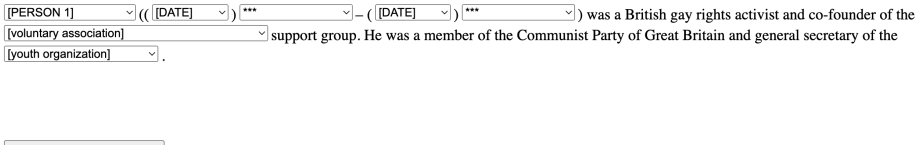

Figure 2: Step 2

Note that there is no one correct solution, as long as the identity of the individual is not disclosed and the replacement choices result in an (as much as possible) informative text.

NB!You have to choose a replacement option. The original string is provided (first option in the drop-down list that cannot be chosen) in order to help choose the most appropriate one. The *Submit and go to next* button can only be clicked if replacements for all the spans have been selected.

Read the text with the selected replacements one last time **(Step 3)**. Make sure that you have chosen replacements for all text spans. Click on *Submit and go to next* to continue with the rest of the texts for this task.

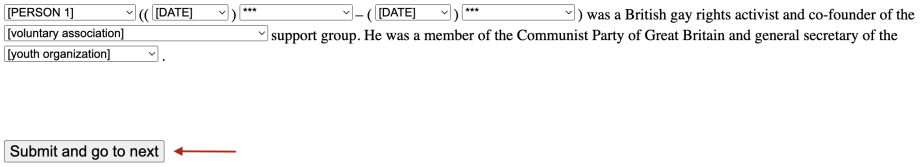

Figure 3: Step 3

A short message will appear on your screen when your assigned number of texts have been annotated.