MSc Artificial Intelligence

Thesis

30 ECTS

The Fragmented Nature of Privacy

A Study on Data Leakage from a Post-Trained BERT Model

Author:
M.S. Verschoor
(2712507)

First Supervisor VU:
D. van den Berg
Second Supervisor VU:
P.J.Mosteiro Romero
First Supervisor Police:
E. Herrewijnen

October 25, 2022

Acknowledgements

For the last eight months, a large part of my life has been dedicated to writing my thesis and finishing my Master's Degree in Artificial Intelligence at the Vrije Universiteit Amsterdam. I could not have done this without the support of several people.

First of all, I'd like to express my thanks to Elize Herrewijnen, who not only made our Monday morning catch-ups in the office 'gezellig', but also contributed majorly in helping me grasp the subject of BERT, the programming, and was always willing to proofread chapters.

In addition, I would like to give thanks to Daan van den Berg, who helped me learn that during this intensive period, there are lessons to be learned that go beyond what the university teaches us.

I am grateful for getting the opportunity to do my thesis internship at the National Policelab AI. Not only has it inspired me in my search for future jobs, it also provided me with exciting insights and opportunities (thanks to Linde Kuijpers for my once-in-a-lifetime visit to Den Bosch).

Lastly, a special thanks to my family and friends, without whose support I could not have written this thesis. Thanks to all who wanted to go for (coffee) walks with me (for 2 hours because the weather was just too nice), or the people who let me vent my frustrations about programming (for the n^{th} time), but most of all the people who were just 'there' when I needed them.

Mincke Verschoor
October 2022

Abstract. The aim of this thesis is to get a better understanding of possible data leakage from sharing language models such as BERT with third parties. XLM-RoBERTa was post-trained on the IMDB dataset, where the five most common attributes were used for further inspection of data leakage. The model was fine-tuned for a Masked Language Modeling task, where 15% of the occurrences of the attributes were replaced with [MASK]. A metric is proposed to quantify data leakage by measuring the model's ability to correctly predict the attribute of the [MASK] in a text. Results diverge from 0.77% to 23.67% data leakage. The conclusions are as follows: attributes with an infinite number of possibilities have less data leakage than attributes with a finite number[1].

Keywords: Data Leakage · Transformers · BERT · Security · NLP.

[1] All Python codes, datasets and outputs are available in the GitHub repository.

Table of Contents

1 Introduction

> "[Blackboxing is] the way scientific and technical work is made invisible by its own success. When a machine runs efficiently, when a matter of fact is settled, one need focus only on its inputs and outputs and not on its internal complexity. Thus, paradoxically, the more science and technology succeed, the more opaque and obscure they become."
> -Bruno Latour, 1999.

Latour, a French sociologist and philosopher, claimed in the late 1990s that a well-performing machine is like a black box. People's focus would lie on the input and output that the machine receives and gives, and would disregard the machine's inner workings. And why would they not? For example, smartphones, Siri, and webshops work accordingly to the user's satisfaction. Clicking 'yes' to the cookies box is just an annoying extra hurdle when surfing the Internet, necessary for the websites to work. However, when given a glimpse inside this black box, people would see a system that collects and sells private user information and (un)consciously regulates behaviour[2].

The question is: if this black box (e.g. a smartphone) collects and sells private user information, why is society not more cautious? Has technology simply become too crucial to everyday life so that we turn a blind eye to its dangers? Or, as Mark Zuckerberg mentioned in 2010, have people become more comfortable with openly sharing information with others, showing that privacy has an evolving "social norm"[13]? The fragmented nature of privacy may not be common knowledge to all, which is why the discussion of privacy concerns in technology should become an increasing fixture in public discourse. First, an example is given that sheds light on the illusion of privacy that people have when using technology's black box.

(1) Imagine a machine learning model that is trained to predict where, when and what kind of burglaries will take place in the province of Utrecht in The Netherlands. This model was trained on a dataset that comprised all burglaries which have taken place between January 2021 and January 2022 and contained, amongst others, names and addresses of the burgled people. Suppose that this model was trained by the Regional Unit of the Police in Utrecht to predict and prevent potential crimes in the province of Utrecht. Now consider the idea that the Regional Unit of Groningen has heard of the success of this model and asks its colleagues to share this model with them. The Regional Unit of Utrecht deliberates whether sharing the model might potentially cause harm.

The focal point of this study revolves around the question, is it safe to share such a model with outsiders? Do we even know what is going on inside this model, or do we regard it merely as a black box, satisfied with the output and

[2] For more information, recommended reading is 'Je hebt wel iets te verbergen' by Maurits Martijn and Dimitri Tokmetzis, 2021 (unfortunately only in Dutch).

not considering possible risks? People who are in this database might not even be aware that the model trained on their private data could be shared with third parties. Are there privacy risks connected to the distribution, and if so, what are those risks? Is there a possibility of data leakage of private information?

To study the privacy concerns of models that are trained on sensitive data, a foundation in Artificial Intelligence (AI) and Natural Language Processing (NLP) needs to be laid. NLP is a subfield of AI concerned with understanding, interpreting, manipulating, and generating human language. However, AI endures difficulties with, amongst other things, the ambiguity of sentences, the use of logical formulas in spoken or written stories, and the variability in which a word can be expressed[4][14]. These challenges cause NLP technology to have difficulty understanding, interpreting, manipulating, and generating human language. Some common examples of NLP applications are automatic text translation, spam filtering, and sentiment analysis[2][10][22]. Nowadays, most of the previously mentioned applications of NLP are conducted via Machine Learning (ML) techniques. In addition to NLP, ML can also be applied to other areas of expertise, including physical sciences, healthcare, and E-commerce[21]. Machine learning can be defined as a subtype of AI, with intended use improving from experience without being explicitly programmed to. In the previously introduced real-world example, ML is used to learn from past experiences and upon that, predict where possible next burglaries will take place.

A game changer within the combined field of ML and NLP is the use of pre-trained transformer models such as Bidirectional Encoder Representations from Transformers (BERT), achieving state-of-the-art results in a plethora of NLP tasks[6]. In the current study, three stages of training BERT are discerned: pre-training, post-training, and fine-tuning. BERT was pre-trained in an unsupervised manner on an enormous amount of unlabelled data and released by Devlin et al.[6]. This model can be downloaded and post-trained on a task- or domain-related dataset, allowing the model to have increased task- and domain-awareness knowledge[16]. The post-trained model can then be fine-tuned to a specific task, for example, a sentiment analysis or translation task. In this research, a BERT model is post-trained on the IMDB Movie Review Dataset[17]. The IMDB dataset is used as a proxy for a private dataset, for example, one that includes burglary information. Subsequently, BERT is fine-tuned on a Masked Language Modeling (MLM) task, which consists of masking words in a sentence and letting the model predict the masked word.

Since the goal of this research is to determine data leakage, it is important to define exactly what 'data leakage' is. The scope of privacy and data leakage is wide-ranging; however, since the objective of this research is more limited, a more narrow definition of data leakage is used. Hereafter, data leakage indicates that masked words are correctly predicted via an MLM task in a trained machine learning model. The model consequently 'leaks' the data. This can be quantified as a percentage of how much of the predicted words are accurately reconstructed by the model.

This study is carried out on behalf of the National Policelab AI in The Netherlands. Especially in a situation where the content of the dataset is considered to be sensitive, for example, addresses and names of burgled people, it can be pragmatic to know how much of the data can be retrieved by adversaries. Suppose that a BERT model trained on the dataset in Example (1) has fallen into adversarial hands. Using MLM, adversaries might reconstruct the masked words and thereby deduce in which streets these people live, what their names are, and what kind of items were stolen. It is practical to know how accurate the model can reconstruct these masked words, i.e., how many of the data points the model can 'leak'. By giving a quantification of how well BERT can predict masked words, measures can be taken to mitigate or maximize information leakage.

This thesis will explore the following research question: "To what extent can data leak from partially masked text with the help of a Masked Language Modelling task of a post-trained BERT model?"

2 Related Work

> "When a secret is revealed, it is the fault of the man who confided it."
> –Jean de La Bruyère, 1688.

The ever-increasing attention towards the possibilities that AI brings about, and the rapid advancement of machine learning models like BERT, also introduces more scrutiny towards its possible shortcomings. Privacy concerns about the usage of statistical models were first raised by Frederikson et al.[9] in 2014 and were later supported by concerns in deep learning models in 2015[8]. The origin of these concerns stemmed from the fact that, although the sensitive datasets on which intelligent models are trained remain confidential, the models themselves are often shared between researchers or other third parties. The question that was raised in Frederikson et al.[9] and which started the privacy debate of machine learning models was: "To what extent do the models themselves leak private information, even if the datasets are not disclosed?"

Previous work by Frederikson et al.[9] has focused on the leakage of private information in a medical setting from publicly available models. While the datasets that contain sensitive information (e.g. patient genotype information) are often kept private, the machine learning models learned from them are made public. Frederikson et al.[9] concluded that even in the absence of the original dataset on which the model was trained, adversaries can infer sensitive information (e.g., reconstruct an individual's genotype) about patients through model inversion attacks. Improper disclosure allows the adversary to determine if someone is a carrier of the Alzheimer's gene, which could make someone a target for crime and/or marketing purposes.

Other work in the medical setting by Nakamura et al.[18] has attempted to deanonymise masked patient information, for example, first and last names, hospitals to which they were admitted, and year of procedures. They state that there

are no guidelines for publishing ML models, as the impact of such models on privacy is unknown to date. They achieved an accuracy of 77.5% in reconstructing complete patient names and complementary diseases. Aside from the fact that unwilling disclosure of a patient's sensitive data can be considered harmful, adversaries can use this leaked information for marketing purposes, targeting vulnerable people (e.g., make a phonebook for patients with leukaemia)[18][20].

Pan et al. [19] demonstrate that an adversary with minimal domain knowledge can deduce sensitive information out of sentence embeddings with an accuracy of 75%. Pan and colleagues studied the amount of data leakage that occurred when an adversary tried to infer whether a certain keyword k was included in an unknown sentence. Similarly to the aforementioned studies, their research was done in the medical domain and concluded that through an adversarial attack, a person's identity, gender, disease type, and birth date can be inferred, given only a small piece of original data. The keyword could be highly sensitive and contain indicators that allow the adversary to infer the locations, diseases, or medical history of a victim. In addition to research in the medical domain, Pan et al. studied data leakage in an airline-related attack. This attack could potentially leak sensitive information such as locations, itinerary, residence, etc. Attacks on both domains (that is, airline and medical), achieved an accuracy of 75% data leakage. Pan et al. stress that this poses a risk to privacy-related domains such as finance, genomics, and health care domains. However, this thesis shows that in addition to the domains mentioned above, adversarial attacks to extract data in the police domain are also a substantial problem.

2.1 The Dangers of MLaaS Companies

In today's society, there is an accumulating presence of ML-as-a-Service (MLaaS) companies, such as Amazon Machine Learning services, Microsoft Azure Machine Learning Studio, and IBM Watson Machine Learning. Aside from these billion-dollar companies, other small-scale MLaaS enterprises are springing up like mushrooms. MlaaS companies offer off-the-shelf, general-purpose machine learning models which can be fine-tuned to a specific demand by uploading one's personal dataset. With an estimated annual growth rate (CAGR) of 39.5% and an expected market growth from $2958.5 million in 2021 to $21803.03 million in 2028, it can be said that business is booming[1]. An illustration of clients for these MLaaS services is given in Example (2) below.

(2) Take for example a healthcare clinic, focused on treating patients with lung cancer, with zero experience in ML and programming. To make the clinic run more efficiently, they turn to an MLaaS company that specialises in detecting small abnormalities in the lungs. After uploading the CT scans of the lungs, as well as additional patient information (age, name, whether they had cancer before, what kind of treatment they received, etc.), they obtain a diagnosis including a proposed treatment plan.

This example shows a correlation with the previously introduced black box, as explained in Example (1). As long as the output of this black box (i.e., the

lung abnormality detection model) is satisfactory, the client might not focus on possible privacy shortcomings that the MLaaS company possibly has. Furthermore, the healthcare clinic might not even know that there is a potential for misuse of their data. After all, their expertise lies within healthcare, and not in machine learning. A malicious MLaaS provider could take advantage of this and gain access to sensitive data that the clinic (and inherently also the clinic's patients) trusted them with. Therefore, can we really place the responsibility for the leakage of sensitive and private data on these gullible clients? This only shows why the applicability of the current research extends beyond the police. By quantifying data leakage within ML models, this study shows how dangerous it can be to share ML models, but also hopefully brings along more scrutiny towards privacy concerns in machine learning.

2.2 A Case-Study with a Random MLaaS Company

A small experiment was conducted to test the reliability of a random MLaaS company. This company, Amberscript, was randomly selected from a list of AI start-ups in The Netherlands[3]. Amberscript[4] provides a tool that automatically turns audio and video into subtitles and text using ML. The only thing the user needs to do is upload their personal data onto the website, after which Amberscript turns this into transcribed audio/video files. This paragraph aims to show the dangers of MLaaS companies; What do they do with your data? Someone who is not acquainted with ML might not know to what extent their privacy is jeopardized when using services like these. This subsection hopefully brings more attention and scrutiny to the possible shortcomings that such, self-proclaimed, "time-saving" tools might have.

Amberscript advertises that it transcribes fast, accurate, and safe (conform the GPDR[5]). They allow interested, possible customers to try out their services by transcribing a video/audio document with a maximum of 10 minutes. A random video, "Liberation Day & staff shortage | The Evening Show with Arjen Lubach"[6] was uploaded to Amberscript to test whether it is actually fast, accurate, and safe. Table 1 describes the first 10 seconds of Arjen Lubach's video, with a transcription through Amberscript and a transcription through automatic YouTube subtitles. Since the original text is in Dutch, a translation is provided through DeepL.

As can be seen from Table 1, the transcription via Amberscript is not completely flawless. In fact, automatic transcription through YouTube yields an even better result. So, while Amberscript advertises that it delivers the transcribed text with the highest accuracy, the actual result still leaves much to be desired. Amberscript keeps its promise for fastness. With the transcription finished in a

[3] A website containing an overview of AI start-ups.
[4] The public website of Amberscript.
[5] General Data Protection Regulation, as described in the GPDR Instruction Manual.
[6] YouTube video: 'Bevrijdingsdag & personeelstekort | De Avondshow met Arjen Lubach (S1)'.

Dutch (Original)	**Text**
Actual Text (NL)	"Welkom bij de Avondshow, mijn naam is Arjen Lubach. Een fijne Bevrijdingsdag allemaal. Het blijft toch leuk om de koning en de koningin uit te zwaaien."
Transcribed by Amberscript (NL)	"'Mijn naam is any lubach en een fijne Bevrijdingsdag allemaal, ja en ik, het blijf toch leuk om de Koning en de Koningin uit te zaaien."
Subtitles via YouTube (NL)	"Welkom bij de Avondshow, mijn naam is Arjen Lubach. Een fijne Bevrijdingsdag allemaal. (gejoel) Het blijft toch leuk om de koning en de koningin uit te zwaaien."
English (Deepl Translation)	**Text**
Actual Text (EN)	"Welcome to the Evening Show, my name is Arjen Lubach. A happy Liberation Day to all of you. It's still nice to see the king and queen off."
Transcribed by Amberscript (EN)	"My name is any lubach and a happy Liberation Day all, yes and I, it still remains fun to Sow out the King and Queen."
Subtitles via YouTube (EN)	"Welcome to the Evening Show, my name is Arjen Lubach. A happy Liberation Day to all of you. (cheering) It's still nice to see the king and queen off."

Table 1: The First 10 Seconds of a Random YouTube Video of Arjen Lubach, Transcribed by Amberscript Versus YouTube's Automatic Transcription. Amberscript's Transcription Lacks Severely in Comparison to YouTube's.

little over ten minutes (for an mp4 of exactly 9 minutes), it is definitely faster than manual transcription. Which takes, according to Amberscript, 5-6 hours for 1 audio hour.

Aside from advertising its accuracy and speed, Amberscript also guarantees its customers' safety and privacy conform to the GPDR. The privacy policy[7] states that uploaded audio is anonymously used as training data for improving the speech-to-text engine, and only after emailing customer service, your data will not be used. Amberscript further specifies in its policy that they use their customers' data 'exclusively' for research purposes, capturing market insights in order to come up with offers that are relevant to their target group, optimising the website performance, improving customer service, and evaluating their products. However, this information is only accessible in the Dutch version of the privacy policy (under the fifth subsection)[8], so English customers do not have access to this information.

Overall, quite a long list to which a customer agrees when using the services of Amberscript. Imagine that this thesis would have contained interviews

[7] The English Privacy Policy of Amberscript.
[8] The Dutch version of the Privacy Policy of Amberscript.

with police employees about data breaches and that Amberscript would have been used to transcribe these interviews. Simply working with Amberscript and accepting the cookies on this website, would have resulted in them using the information from the interviews for "research purposes" (whatever this may be), and amongst other things, using the data to capture "market insights" and use my own data to offer me relevant offers.

However, the sword cuts in both ways. While clients of MLaaS companies should be concerned about their data being used for nefarious reasons, MLaaS companies are also wary of being totally transparent in how exactly they handle the data that are entrusted to them. Their MLaaS algorithms are usually black boxes; the client knows nothing about the inside workings of the algorithms, how the model is trained, and how their data are being processed. An answer to the distrust of both sides could be Chiron, a system for privacy-preserving MLaaS models[12]. Developed by Hunt et al., this system's main purpose is to protect clients of MLaaS companies from potential malicious providers. However, Chiron also does not let the MLaaS client peak under the hood of the MLaaS company's algorithms. This way, the privacy of both sides is protected.

However, does its efficiency and acceptable transcription balance out the lack of consideration for one's privacy? That is something for the user to decide. However, might one consider utilising the service of an MLaaS company, it does not hurt to inspect the privacy policy to check exactly how the training data are being used.

3 Methodology

The current section will introduce some preliminary information, such as the concept of transformers, attention, and different attacks. These notions lay the foundation for understanding how the results of this study are acquired. After the groundwork is laid, the approach for generating results will be discussed.

3.1 Transformers and Attention

Previous approaches to solving NLP tasks generally revolved around neural networks (NN), where varying neural networks have been proposed for varying tasks. These techniques have laid down the foundation of avant-garde research, achieving great results in tasks such as language modelling and machine translation[24][3]. However, NN algorithms show a disability in dealing with long-range dependencies[7]. Unlike computers, humans can naturally identify syntactic dependencies in sentences. This is illustrated by the sentence in Example (3).

(3) An example sentence with long-range word dependencies.

a. "The U.S. presidential race isn't only drawing attention and controversy in the United States – it's being closely watched across the globe. But what does the rest of the world think about a campaign

> that has already thrown up one surprise after another? CNN asked 10 journalists for their take on the race so far, and what their country might be hoping for in America's next [MASK]. We'll also be checking in again with some of them as the race continues." (Source: Dieng et al.[7].)

Computers have a greater difficulty capturing long-range dependencies in sentences. In sentence (3)a, humans should be able to connect 'America's next [MASK]' to the U.S. presidential race; however, for computers this is not as straightforward. Their algorithms could just as easily predict 'America's Next Top Model,' or 'America's next Generation' as 'America's next President'. Long-range dependencies are important when answering open-domain questions, sentiment analysis, and text summarisation, among other uses. A solution for this problem was the transformer model, as introduced by Devlin et al.[6].

Devlin et al.[6] proposed adding an attention layer to the neural network structure, which has proven to be revolutionary for the NLP tasks mentioned above, and causes this transformer model to surpass previous NN approaches. Previous approaches have difficulty handling input with longer sequence lengths due to computational memory constraints. This is also called the Bottleneck Problem or Vanishing Gradient Problem[11], since the algorithm tends to 'forget' information from earlier time steps. Instead of handling input sequentially, the attention mechanism can model dependencies disregarding the distance in the sentence, thereby inherently manoeuvring around the sequential computation problem.

Take for example an ML model that is trained to translate English sentences into Romanian sentences. Instead of translating each word in the sentence separately (and sequentially), it takes the whole sentence into account and assigns weights to the words that contribute the most to the meaning of that sentence. These weights are also called embeddings, which are saved in real-numbered vectors. Every word has a unique word embedding, which can capture not only the context of a word in a sentence but also the syntactic and semantic similarity, the relation to other words, etc. With the help of attention, the word 'date' in the sentence 'They went on two very successful dates,' has a different word embedding than the word 'date' in the sentence 'Dried dates are my favourite snack'. The model uses these word embeddings to translate the English sentence into a Romanian sentence. This approach has been shown to generate better translations than the approach without the attention layer.

In the current study, attention is applied as a way of improving the ability to predict masked words. When looking at the entire sentence, the model can reconstruct the masked words better than by looking at the words sequentially. As a result, the Vanishing Gradient Problem is no longer an issue when working with the attention mechanism.

3.2 BERT

Bidirectional Encoder Representations from Transformers (BERT)[6] was developed at the end of 2018 by Google researchers and is considered a breakthrough in the NLP community. One of its greatest strengths is the fact that it is pre-trained on an unlabelled corpus without a real training objective. The more data used to train the model, the better the results. However, until the development of BERT, it was necessary to use human-annotated labelled data. The arrival of BERT changed all of this. Through unsupervised learning, BERT can accurately dissect the context and meaning of words from a large amount of unlabeled data. After the time-consuming pre-training, BERT is a general-purpose model that can quickly and inexpensively be post-trained and fine-tuned on smaller task-specific datasets, resulting in, e.g., a sentiment analysis model or a spam filtering model.

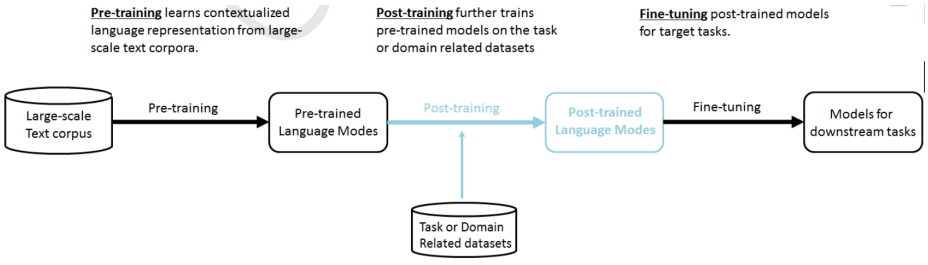


Fig. 1: An Illustration of the Three Stages of Training a BERT model. Pre-training Results in Language Models like BERT, via Post-training these Language Models are Specified on a Task- or Domain-related Dataset. During Fine-tuning these Post-trained Models are Finalised into Models for Downstream NLP Tasks (Image from Liu et al.[16]).

After pre-training two more stages can be distinguished: fine-tuning and post-training. Post-training on the task-related dataset before fine-tuning, increases task-awareness knowledge within the pre-trained language model, as well as reducing bias[16]. This approach outperforms the existing BERT benchmarks, according to the results required by Liu et al.[16]. Therefore, this study first post-trains the pre-trained BERT model on the IMDB dataset, and after that, the model can be fine-tuned for any downstream NLP task. The three stages are visualised in Figure 1. Figure 2 depicts the reconstruction of a masked word, with and without post-training. As can be seen from the figure, post-training results in the prediction of the masked word to be specified on the movie domain.

Fine-tuning BERT on a specific downstream NLP task can result in, e.g., a sentiment analysis model, an emotion detection model, or a text summarization model. The post-trained model is fine-tuned for a Masked Language Modelling (MLM) task. This task is used to predict how much data leakage occurs, by randomly masking about 15% of the words in the test dataset, after which the model tries to predict the masks. Looking back on the definition of data leakage, the model 'leaks' data when the post-trained model correctly predicts the

masked word. If the BERT model were post-trained on the burglary dataset, as introduced in Example (1), the model could, for example, leak locations, names, or stolen items.

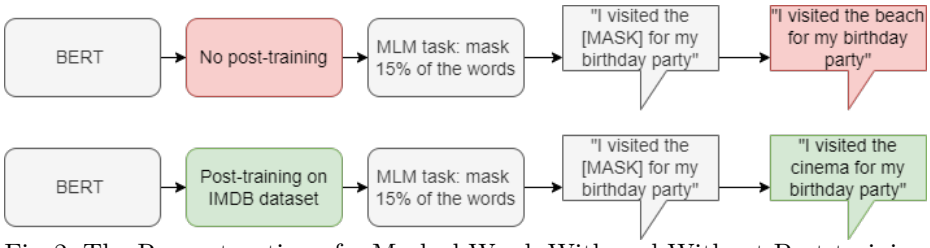


Fig. 2: The Reconstruction of a Masked Word, With and Without Post-training on the IMDB Dataset. Post-training Yields a Model Specified on a Task- or Domain-related Datset, Causing the Post-trained Model to Surpass Models Without Post-training.

Over time, different versions of BERT have emerged, optimised for domain-specific datasets, or trained in different languages. For instance, BERTje[25], a customised BERT model pre-trained on Dutch text, or XLM-roBERTa[15], a multilingual BERT model that is pre-trained on 100 different languages.

3.3 Attacks

In order to let the post-trained ML model 'leak' private information, three broad types of attacks can be distinguished[23]. The goal of the adversary is to leak confidential information from the post-trained model and depending on what kind of information the adversary wants, different kinds of attacks would be more suitable. If an adversary gains access to a leaked model, they only acquire access to the embeddings, that is, the real-numbered vectors. The aim is to revert the embeddings to actual words, which allows the adversary to gain access to the leaked information.

Embedding Inversion Attack In this attack, the adversary aims to (partially) revert the embedding to the original text. Considering a situation where someone wants to get insight into data such as personal messages, an embedding inversion attack would be most fitting.

Membership Inference Attack Through a membership inference attack, the adversary aims to leak information about training data. Specifically, the attacker wants to find out if a data point is in the training set of a model, e.g. does 'London' appear in the training dataset?

Sensitive Attribute Inference Attack During a sensitive attribute inference attack, the adversary tries to infer if a masked word x is of a specific category. These categories are called attributes, such as locations, dates, and events (see Table

2 for an overview of all attributes). In this study, a sensitive attribute inference attack is carried out. Imagine that the burglary model (Example (1), as discussed in the Introduction) would be shared with third parties with (un)known malicious intent. This model consists of embeddings that contain extensive semantic information about victims of burglary, quite sensitive information. An adversary would want to find out if the information that the model contains is of any interest; if the model encompasses locations, works of art, or names. Using an MLM task, the model predicts the masks in text, and consequently lets the adversary discover via the predicted words if the model is of any interest.

4 Experimental Setup

4.1 Data

Since the results should be publicly available in consideration of the grading, the IMDB Movie Review Dataset is used as a proxy for a 'real', sensitive dataset. This dataset was published by Maas et al.[17], and is originally intended for sentiment classification. To classify the different attributes in the sentences, SpaCy's Named Entity Recognition (NER)[9] is performed on the IMDB dataset. An example of a review and the corresponding labelling is shown in Figure 3. As can be seen in the figure, SpaCy's labelling is not flawless; 'Oz,' a nickname for 'The Oswald Maximum Security State Penitentary' was only once correctly labelled as ORG (organisation) out of the six times it was mentioned. Clearly, SpaCy's attribute labelling still leaves much to be desired. SpaCy themselves claim to have an accuracy of 85% in labelling named entities[10]. Table 2 gives an overview of the possible different attributes and the average occurrence of this attribute per review. As can be read from the table, the five most common attributes are PERSON, ORG, CARDINAL, GPE and DATE. These are the five attributes that will be considered as most important, and hence will be used for this study.

4.2 Data Preparation

The IMDB dataset contains 50.000 movie reviews, a division of 80/20 was used for training and testing respectively. The objective of the MLM task is to let the model predict masked words. To create a test dataset for the MLM task, 15% of the occurrences of the desired attribute were masked. For example, the test dataset for the PERSON attribute is created by masking every seventh occurrence ($\approx 14.3\%$) of a PERSON attribute. For instance, if the word 'Shakespeare' in the sentence "He gave Shakespeare a handshake," is the seventh occurrence of a PERSON in the dataset, then 'Shakespeare' is replaced by [MASK]. This results in the sentence: "He gave [MASK] a handshake". In this manner, five test datasets were created for the most common attributes, these can be found in the

[9] Documentation for SpaCy's NER.
[10] As can be read from SpaCY's NER Benchmarks.

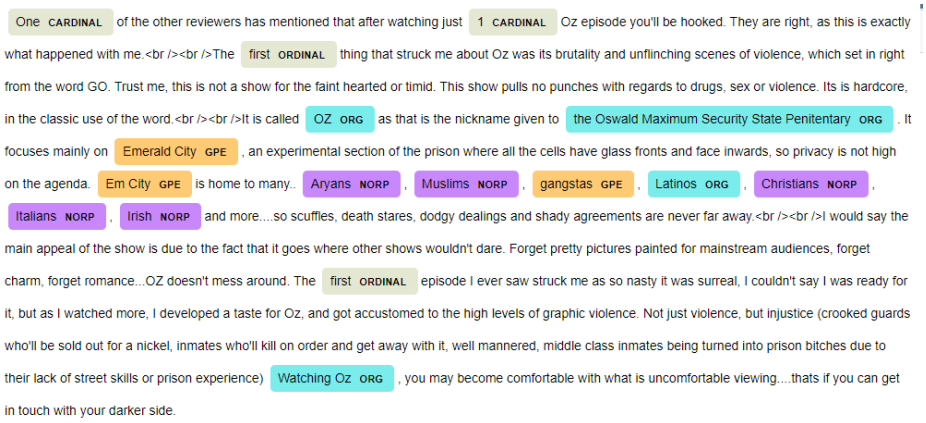


Fig. 3: A Visualization of the Labelling of Different Attributes in a Random Movie Review Sentence by SpaCy. Not All Attributes are Recognised.

Type	Description	Example	Avg Occ
PERSON	People, including fictional	Winnie the Pooh	4.38
ORG	Companies, agencies, institutions, etc.	Facebook	1.81
CARDINAL	Numerals that don't fall under another type	12	1.43
GPE	Countries, cities, etc.	Utrecht	0.89
DATE	Absolute or relative dates or periods	Yesterday	0.86
NORP	Nationalities, religious/political groups	The Flemish	0.65
WORK_OF_ART	Titles of books/songs, etc.	Gulliver's Travels	0.51
ORDINAL	"First" or "Seventh",etc.	"First"	0.39
TIME	Times smaller than a day	30 seconds	0.27
LOC	Non-GPE locations, e.g. lakes, mountains.	The 'Randstad'	0.14
FAC	Buildings, airports, highways, etc.	The London Bridge	0.09
PRODUCT	Objects, vehicles, foods, etc.	Formula 1	0.09
EVENT	Named battles/sports events, etc.	Storm Eunice	0.06
MONEY	Monetary values	1 Million Euros	0.05
LANGUAGE	Any named language	Japanese	0.03
LAW	Named documents made into laws	Fifth Amendment	0.02
QUANTITY	Measurements, e.g. weight or distance	2 Gallons	0.02
PERCENT	Percentages	40%	0.01

Table 2: An Overview of Possible Different Attributes in the IMDB Dataset, and the Average Occurrence per Review of the Attribute.

GitHub repo[11]. To get some more insight into the IMDB datasets, figures for the ten most occurring words of the DATE, CARDINAL, PERSON, GPE, and ORG test datasets were created, respectively, in Figures 5a, 6a, 7a, 8a, 9a[12]. In addition to the test datasets for the five most common attributes, a test dataset

[11] All datasets can be found under 'datasets', e.g., 'DATE_test_dataset.txt'.
[12] All images can also be found in the GitHub repo, under 'imgs', e.g., 'DATE_test.jpg'.

was created for a random baseline. This was done by collecting every occurrence of the five most common attributes and masking every seventh word.

4.3 Attack Model

To find out which transformer model is most fitted for the task at hand, a comparison between models needs to be made. However, it is challenging to compare different transformer models, since training is often done on private datasets of altering sizes and aimed at different tasks. Liu et al.[15] found through a replication study of Devlin et al.[6] that the original BERT model was undertrained, and proposed modifications in order to surpass BERT's performance. They called this modified model RoBERTa, Robustly Optimized BERT Approach. Pan et al.[19] compared the data leakage of different language models (amongst others: GPT-2, BERT, XL, ERNIE, RoBERTa) for a study where they tried to reconstruct citizen ids. Out of all compared models, they found RoBERTa to be the most robust; RoBERTa showed approximately 50% less privacy risk than BERT. The current study uses the XLM-RoBERTa base model. XLM-RoBERTa-base was released by Conneau et al.[5] and was pre-trained on a dataset with 100 different languages, the model systematically outperformed the original BERT model as released by Devlin et al.[6]. However, ultimately this model was chosen because it was the only non-Dutch model available in the offline police environment.

4.4 Metrics to Quantify Data Leakage

The XLM-RoBERTa model is post-trained on the IMDB dataset, after which it is fine-tuned for an MLM task on six test datasets. The model predicts the [MASK] in each of the test datasets, and returns as output a list of all predicted masks[13]. Subsequently, the accuracy of the predicted words is calculated. The accuracy of the attack is determined as follows: If the model correctly predicts 28 attributes of the masked words, of the 300 masks, then the accuracy is (28/300)*100 $\approx$ 9.33%. Thus, the results of this study are quantified by the following formula for accuracy: (number of correctly predicted attributes/number of all masks)*100.

5 Results

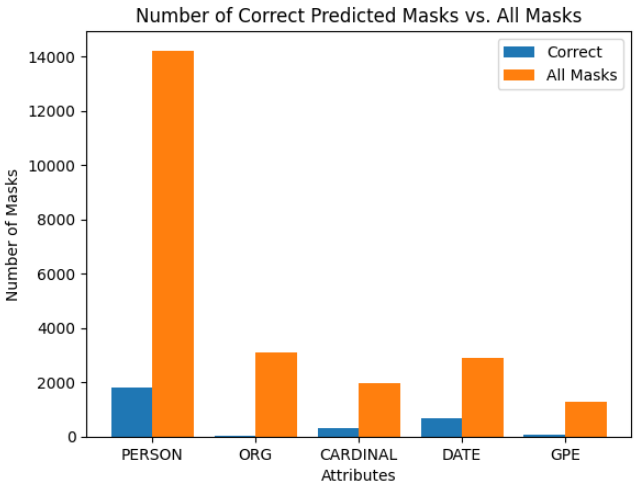
Table 3 shows the data leakage results achieved by the post-trained BERT model, which are visualised in Figure 4a. As can be seen in Table 3, the attribute that is most likely to leak information is the attribute DATE, followed by CARDINAL, PERSON, GPE, and the least likely to leak information is the attribute ORG.

Table 4 reports the findings of how well the model predicted masks with respect to other attributes. As can be seen in the table, the TIME attribute has the highest data leakage, with an accuracy of 18.54%, while PRODUCT, WORK_OF_ART, and LANGUAGE all have an accuracy of 0%. Table 5 presents

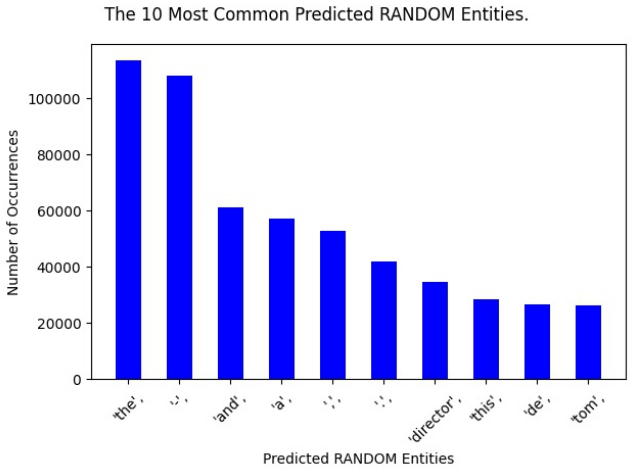
[13] The code to predict masks can be found under 'python_code/predict_dataset.ipynb'.

Attribute	Predicted as Correct Attribute	Number of Masks	Accuracy
DATE	687	2,903	23.67%
CARDINAL	307	1,966	15.61%
PERSON	1,818	14,213	12.79%
GPE	62	1,278	4.85%
ORG	24	3,108	0.77%

Table 3: Reported Results of the Predictions of the Post-trained XLM-RoBERTa Model on the Masked Test Datasets of the 5 Most Common Attributes (DATE, CARDINAL, PERSON, GPE, and ORG). Attribute that is Least Likely to Leak Information is ORG, Most Likely is the DATE Attribute.



(a) Nr. of Correctly Predicted Masks vs. All Masks.



(b) 10 Most Common Predicted Words, RANDOM Dataset.

Fig. 4: Left: a Comparison of the Number of Correct vs. All Masks for the Five Most Common Attributes. Right: 10 Most Common Words In the Predicted RANDOM Dataset.

the results of the random baseline, which specify that the PERSON attribute is the most likely to be predicted, followed by DATE, CARDINAL, ORG, and the least likely to be successfully predicted is the GPE attribute. Figures 5, 6, 7, 8, and 9 illustrate a comparison between the 10 most common named entities in the test datasets and in the predicted datasets of the DATE, CARDINAL, PERSON, GPE, and ORG attributes[14]. As specified by the figures, the 10 most common entities are not identical for both datasets. Taking a look at the DATE attribute, one can see that there are some illogical words in the predicted dataset (see Figure 5b) like 'the' and 'a', and that instead of predicting a word for the [MASK], the model sometimes predicts interpunction. Figure 4b shows the 10 most common predicted words for the masks in the RANDOM test dataset. In

[14] The images are also available online in the Github repo under 'imgs'.

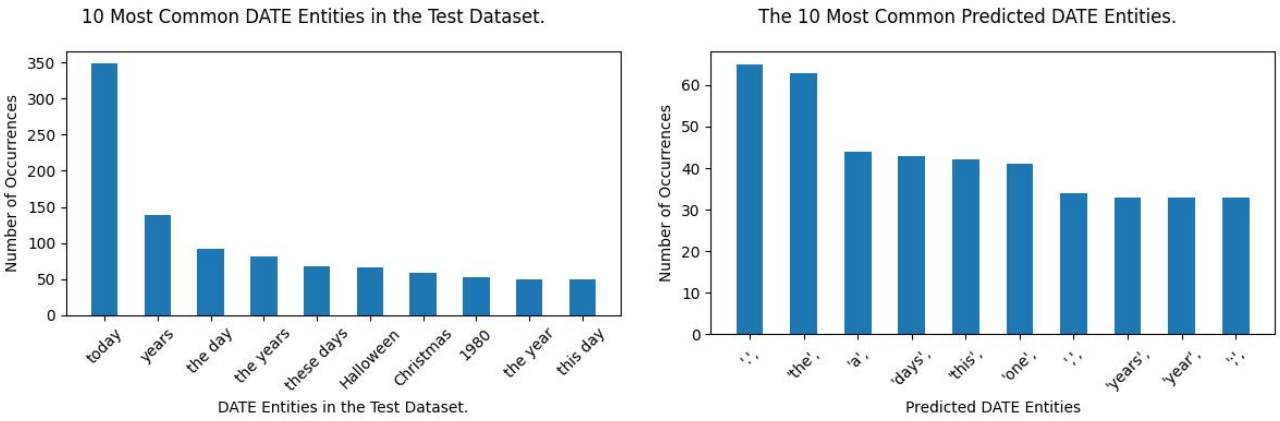
descending order, the most common words are: [‘the’, ‘-’, ‘and’, ‘,’, ‘.’, ‘director’, ‘this’, ‘de’, ‘tom’].

Attribute	Number of Predicted Attributes	Number of Masks	Accuracy
TIME	219	1,181	18.54%
MONEY	25	164	15.24%
ORDINAL	82	602	13.62%
LOC	4	280	1.43%
EVENT	3	241	1.24%
FAC	2	234	0.85%
PRODUCT	0	248	0%
WORK_OF_ART	0	2,609	0%
LANGUAGE	0	55	0%

Table 4: Results of the Predictions of the Post-trained XLM-RoBERTa Model on the Masked Test Dataset of the Other Attributes. The TIME Attribute is Most Likely to Leak Data, while the PRODUCT, WORK_OF_ART and LANGUAGE Attributes are the Least Likely to Leak Data.

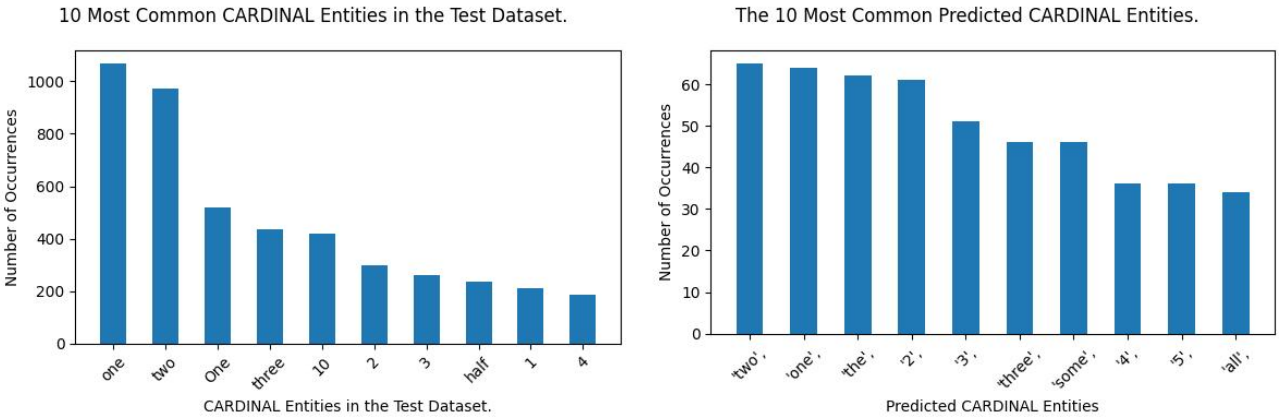
Attribute	Number of Predicted Attributes	Number of Masks	Accuracy
PERSON	1,091	24,716	4.41%
DATE	472	24,716	1.91%
CARDINAL	384	24,716	1.55%
ORG	310	24,716	1.25%
GPE	29	24,716	0.12%

Table 5: Results of the Predictions of the Post-trained XLM-RoBERTa Model on the RANDOM Test Dataset.



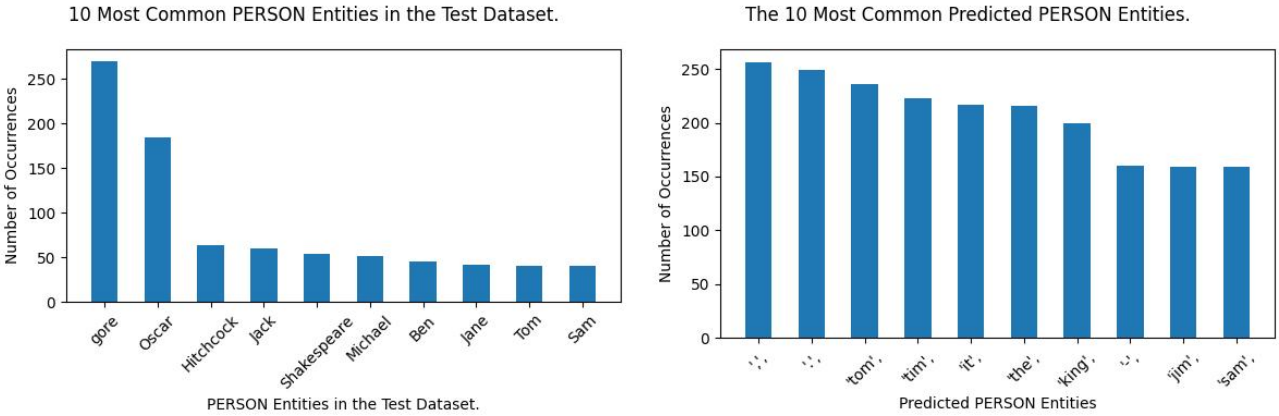
(a) Test Dataset. (b) Predicted Dataset.

Fig. 5: A Comparison of the 10 Most Common Entities in the DATE dataset, Test versus Predicted.



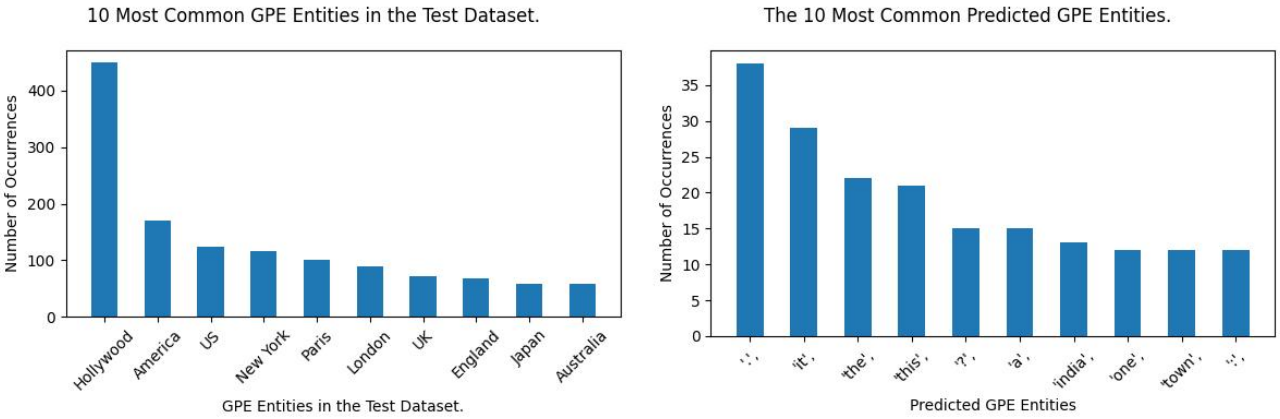
(a) Test Dataset. (b) Predicted Dataset.

Fig. 6: A Comparison of the 10 Most Common Entities in the CARDINAL dataset, Test versus Predicted.



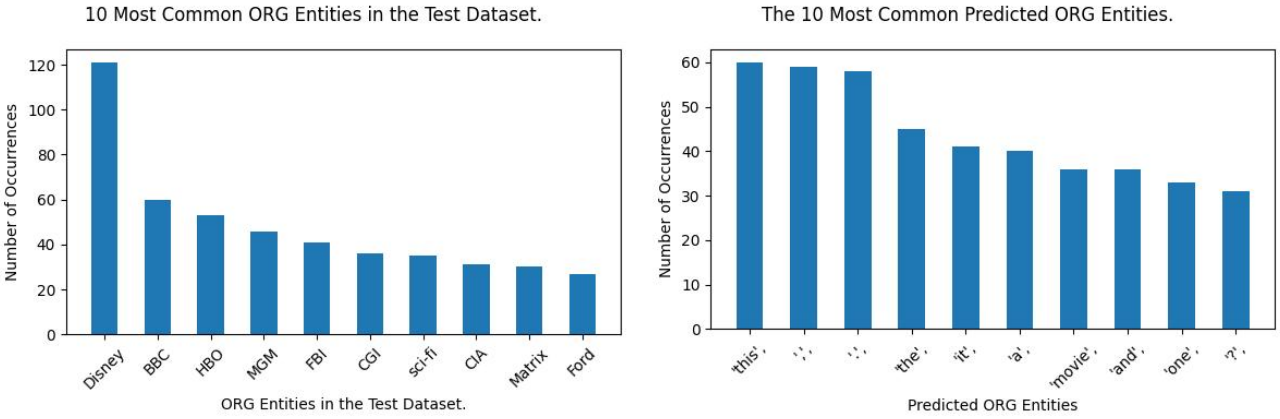
(a) Test Dataset. (b) Predicted Dataset.

Fig. 7: A Comparison of the 10 Most Common Entities in the PERSON dataset, Test versus Predicted.



(a) Test Dataset. (b) Predicted Dataset

Fig. 8: A Comparison of the 10 Most Common Entities in the GPE dataset, Test versus Predicted.



(a) Test Dataset. (b) Predicted Dataset.

Fig. 9: A Comparison of the 10 Most Common Entities in the ORG dataset, Test versus Predicted.

The data (number of correctly predicted masks and the total number of masks) do not follow a normal distribution. To determine whether the results are significant, we have to look at non-parametric tests since the data are not normal. Interest is in finding out if there is a relation between the number of correctly predicted masks and the total number of masks (in Table 3, columns two and three). Spearman's Rank Correlation Test is used to determine the extent to which these two columns correlate. The output of Spearman's test can take a value from +1 to -1, where +1 means a perfect positive correlation, 0

means no correlation, and -1 means a perfect negative correlation.
The following null hypothesis is tested:

- H_0: $\rho = 0$, i.e. the correlation between the number of correctly predicted masks and the total number of masks is 0.

- H_1:$\rho \neq 0$, i.e. the correlation between the number of correctly predicted masks and the total number of masks is not 0.

The p-value is 0.02799 and the ρ is 0.58, which allows us to reject the null hypothesis and accept the alternative hypothesis that there is a correlation. In fact, ρ is 0.58, indicating a fairly strong positive correlation between the columns. This means that when the total amount of masks goes up (e.g., the PERSON attribute occurs more often than the GPE attribute), the total amount of correctly predicted masks also goes up. This is also expected.

6 Discussion

In this study, a closer look is taken at the amount of data leakage that is achieved during a sensitive attribute attack on a post-trained BERT model. Contrary to previous work (that is, Pan et al.[19] and Nakamura et al.[18]), less data are leaked in a post-trained BERT model according to Table 3. With results diverging from 0.77% to 23.67%, one can safely conclude that the amount of data leakage is significantly lower than the results acquired by Pan et al. or Nakamura et al. (respectively, 75% and 77.5%). Looking at the outcomes in Table 3, the results suggest that when a model is confronted with the [MASK] in a sentence, it is easier to deduce from the context that the mask is a DATE than an ORG. Take for example the following two sentences, shown in Example (4), randomly chosen from the DATE and ORG test dataset.

(4) Two sentences were randomly chosen from the DATE and ORG test datasets. The masks represent respectively "9" and "Lubitsch".
   a. "Ever since I was [MASK] years old (I am now 15)."
   b. "Rather than substituting either of his main stars, [MASK] decided to postpone production, in the meantime directing Greta Garbo in 'Ninotchka (1939)."

First of all, we can conclude that the [MASK] in (4)b is incorrectly placed. Lubitsch is a name, and therefore falls into the PERSON category and not into the ORG category. This shows that SpaCy's NER model is not flawless in deducing named entities (SpaCy themselves claim to have an accuracy of 85.5% in labelling named entities[15]). Leaving aside the fact that the [MASK] in (4)b has a wrong placement, the list of possible words in (4)b is also longer. In (4)a, one can deduce from the context that the logical word would be a number between [0-14]. Not a lot of possibilities. However, for the [MASK] in (4)b, the list

[15] SpaCy's NER Benchmarks, https://spacy.io/usage/facts-figures

of possibilities is endless, for example, all companies, agencies and institutions (which is an infinite list, with more organisations opening up every day). And, aside from all organisations, it seems as though the masks in the ORG test dataset also have a possibility of being filled up by a PERSON attribute. This means that in addition to the infinite list of all organisations, the infinite list of all persons is also added to the list of possible words for the [MASK] in (4)b. Not quite the 15 possibilities that the [MASK] in (4)a has. Relating this to Table 3, a hypothesis can be proposed that this is the reason why the DATE attribute has more data leakage than other attributes; the DATE masks are simply easier to deduce from the context of the sentence than the other attributes.

This theory is also supported by the findings in Table 4, which show the predictions of the model with other masked attributes. As can be seen in Table 4, attributes that are bounded to a finite amount of possibilities (e.g., words that indicate time, money or ordinals) have high accuracy. Vice versa, attributes with more possibilities (e.g., locations, events, and products all have endless options) have shockingly low accuracy. Another contributing factor to the low accuracy scores could be the fact that a PRODUCT [MASK] is more easily substitutable than a TIME [MASK]. For example, in the place of a PRODUCT [MASK], the model could perhaps just as easily predict a WORK_OF_ART. However, in place of a TIME [MASK], very few other words are logical (see also the DATE example in Example (4)a). There is one clear exception: the LANGUAGE attribute. Although it has a clear margin of possible words, it has an accuracy of 0%. This could be because the LANGUAGE attribute is also the least occurring attribute. Therefore, the model might not have had the chance to post-train enough on this attribute.

It is also interesting that the random baseline has such low accuracy. When presented with the [MASK] in the random test dataset, the model is not accurate in predicting the correct attribute. Looking at Figure 4b, which depicts the 10 most common predicted words; one can see that only the words ‘director’ and ‘tom’ are words with meaning, the rest are stopwords and interpunction. This observation is not coincidental, the occurrence of stopwords and interpunction can also be seen in Figures 5b, 6b, 7b, 8b, and9b. These incorrect predictions lower the accuracies of the attributes. Liu et al.[15] concluded that the more epochs the model was post-trained, the more improvement could be seen in the predictions. A logical follow-up study would therefore be to post-train the model for more than one epoch, which would hopefully improve predictions. However, if this does not succeed, a more radical approach of removing all stopwords and interpunctions from the IMDB dataset could be next. After this, the BERT model would be post-trained on the altered dataset, and accuracies will hopefully improve.

The occurrence of stopwords and interpunction is illustrated in Example sentence (5). SpaCy recognised three named entities, ‘Queen Elizabeth II’, ‘St. George’s Chapel’, and ‘Prince Philip’ (again a flaw in detecting named entities by SpaCy by neglecting to detect Windsor Castle). All three masks can be

replaced by stopwords or interpunction, and after replacing, they still form a correct sentence.

(5) An example sentence in which masks can be replaced with stop words and interpunction and still form a correct sentence.
   a. "Queen Elizabeth II was buried in St. George's Chapel at Windsor Castle, next to her husband, Prince Philip."
   b. " [MASK] was buried in [MASK] at Windsor Castle, next to her husband, [MASK]."
   c. "**And** was buried in **this** at Windsor Castle, next to her husband, **.**"

A possible explanation for the abundance of stopwords is that BERT itself is a model that focuses on the continuity of the sentence. As a consequence, BERT is more inclined to fill the [MASK] with a stopword rather than a content word. However, no literature was found that can support this claim. It could be interesting to conduct further research into interpunction substitution. If the [MASK] is placed at the end of a sentence, will it be more often substituted with interpunction than if the [MASK] occurs in the middle of a sentence?

One thing that might have influenced the accuracy is the way the calculations were performed. The predicted masks were returned in a list that did not contain the sentences that surrounded the mask. Therefore, the sentence "Legally Blonde's Elle [MASK] wasn't supposed to go to Harvard!" ([MASK] replaced the surname 'Woods'), returned a list with just ['Woods'] (presuming that the model correctly predicted 'Woods' as the [MASK]). Just the word 'Woods' is not recognised by SpaCy as a PERSON, just the word 'Elle' is also not recognised by SpaCy as a PERSON attribute. However, the combination 'Elle Woods', is recognised as a PERSON. This happens to more words; 'Johnny' is recognised as a PERSON, 'Depp' is not, and the combination 'Johnny Depp' also isn't recognised as a PERSON. 'Amalia' is not recognised as a PERSON, 'van Oranje' on the other hand is recognised as a PERSON, but the combination isn't. However, delving into this subject would be a whole other thesis research. An informal study[16] compared SpaCy performance with Google's NLP AI on tasks such as the recognition of named entities. They concluded that Google's NLP AI outperformed SpaCy in labelling brands and shops (ORG and PRODUCT attributes). It would be interesting to see if accuracies would improve with this other method of recognising named entities. However, since it is an API, this could not work in the secure environment of the police.

In this study an XLM-RoBERTa model was used, which is pre-trained on 100 different languages. It excels in tasks where multiple languages need to be recognised, however in this research the model only needs to work with English language. For future research it could therefore be interesting to see if there is less or more data leakage when a sensitive attribute attack is executed on a BERT model specifically for English language. Previous research into this field by Pan

[16] Source of the claim that Google's NLP AI outperforms SpaCy's NER.

et al., [19] concluded that there was a difference between language models in robustness and, therefore, in their ability to contain data leakage. On average, Pan and colleagues' reported accuracy when trying to reverse-engineer masks was 75%. The research by Pan et al.[19] is not completely similar to the research methodology of this paper. Pan studied data leakage by correctly predicting if a keyword k is included in an unknown sentence, this research studies the data leakage via correctly predicting the right attribute in a sentence. The accuracy for correctly predicting a keyword was 75% according to Pan et al. Also doing research in the medical domain are Nakamura and colleagues[18], who achieved an accuracy of 77.5% when trying to de-anonymize masked patient information. Although both researches achieved high accuracy, this study reports a maximum accuracy of only 23%. This could be traced back to multiple reasons: a different domain, different attack models, different datasets, a shorter period of research time, etc. The fact is that due to the many differences in the research setup, a real comparison cannot be truly drawn.

What does all this mean in relation to the police? We hope to answer this question by linking the results of this study to the burglary example, introduced in Example (1). When this predictive burglary model is shared with third parties, what kind of data are most probable to leak? In this scenario, the adversary attains access to the model and lets the model predict words on a masked test dataset. If the masked word was originally a DATE attribute, the model predicts in 23,67% of the cases also a DATE. If the masked word was originally a CARDINAL, it predicts in 15.61% of the cases that the [MASK] is a CARDINAL, etc. with the other attributes (see Table 3). Example (6) illustrates a sentence from a fictitious test dataset.

(6) The adversarial attack is illustrated by the following mock sentence in a, the masked variant in b, and the predicted sentence in c:

a. "Three years ago, on June 15, 2019, the elderly couple Mr. and Mrs. Johnson were robbed. Four items were stolen from their home on Domplein 29 in Utrecht: a television, two valuable necklaces (family heirlooms), and a Monet painting. All stolen items accumulated to a total of 15.6 million dollars.

b. "Three [MASK] ago, on June 15, 2019, the elderly couple Mr. and Mrs. Johnson were robbed. Four items were stolen from their home on Domplein 29 in [MASK]: a television, two valuable necklaces (family heirlooms), and a Monet [MASK]. All stolen items accumulated to a total of 15.6 million dollars.

c. "Three **years** ago, on June 15, 2019, the elderly couple Mr. and Mrs. Johnson were robbed. Four items were stolen from their home on Domplein 29 in **India**: a television, two valuable necklaces (family heirlooms), and a Monet **!**. All stolen items accumulated to a total of 15.6 million dollars.

The generated sentence in (6)c illustrates the three different options for predicting the [MASK]. In the place of the first [MASK], the model predicted the right

attribute and the right word (years). The second [MASK] was replaced with the correct attribute, but with the wrong word (India instead of Utrecht). In the place of the third [MASK], an exclamation mark was predicted; both not the correct word and the correct attribute. The second prediction in (6)c also emphasizes one of the weaker links of the characteristics of the sensitive attribute attacks. Even if a predicted word is not the original word, as long as it is the correct attribute, it counts as a success for the accuracy calculations. This means that 'India', even though it is not the correct city (not even the correct country), is considered a correct prediction since it is the correct attribute.

The adversary thus might have access to the model, and make predictions on a masked dataset, but this does not mean that the predictions contain the right information. Yes, in 23.67% of the cases, if a word is masked and it is a DATE, the predicted word is also a DATE. However, just because the predicted word is a DATE, does not mean that the model predicted the right word. The 'real' data leakage of actual information that is valuable to the adversary is therefore much lower than the calculated 23.67%. However, that would be considered an embedding inversion attack, and not a sensitive attribute inference attack, which was the purpose of this study.

An oft-proposed solution to mitigate information leakage is differential privacy (DP). Frederikson et al.[8] studied DP as a way of mitigating data leakage and found a trade-off between the utility of an ML model and protecting privacy. It could be interesting to check if implementing differential privacy is effective in reducing data leakage in a post-trained BERT model during a sensitive attribute attack, and if by the using DP, concessions are made in the utility of the model.

7 Conclusion

This work attempts to quantify the amount of data leakage during an in-depth analysis of a sensitive attribute attack on a post-trained BERT model. An XLM-RoBERTa model was post-trained on the IMDB dataset, after which the five most occurring attributes were identified as DATE, PERSON, GPE, CARDINAL, and ORG. After post-training, the model was fine-tuned for a Masked Language Modeling (MLM) task, where 15% of the occurrences of the attributes were masked. The goal of this study was to let the post-trained model predict the [MASK] in the test datasets, the model 'leaks' data when it predicts the right attribute in place of the [MASK]. Accuracies from 0.77% to 23.67% are reported (respectively for the ORG and DATE attributes), these accuracies are significantly lower than previous studies report.

In summary, the following insights were acquired:

– Attributes that have a finite number of possibilities (for example, dates, money, and times) have more data leakage than attributes with an infinite amount of possibilities (for example, works of art, persons or objects).

- The model often predicted stopwords and interpunction instead of 'normal' words. A proposed theory is that BERT is more focused on the continuity of the sentence, rather than filling in the [MASK] with a content word.
- Perhaps accuracies could be improved by using a different Named Entity Recognition technique; Google's NLP AI has been shown to outperform SpaCy's NER.
- The fact that the maximum accuracy is only 23.67% compared to a maximum of 77.5% in other studies (Nakamura et al.[18]) could be due to multiple reasons. A different domain, different attack models, different datasets, a shorter period of research time, etc. A true comparison is hard to draw due to the differences.

In conclusion, the sensitive attribute attack in this study achieved less data leakage than in previous research. However, future applications of general-purpose language models like BERT should be mindful towards the possible data leakage that occurs when sharing the model with third parties, especially in privacy-critical domains like healthcare, crime, and finance where training data is considered to be sensitive. For now, let us hope that the more transformers like BERT are used, the more attention is also drawn towards its shortcomings in privacy. Perhaps some basic guidelines will be formulated about sharing the model with outsiders in order to protect everyone's privacy (especially of those who can not vouch for themselves). Let us make the fragmented nature of BERT's privacy in the future a little less fragmented, and a little more safe.

References

1. Machine learning as a service market. Vantage Market Research (2021), https://www.vantagemarketresearch.com/industry-report/machine-learning-as-a-service-market-market-1350
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
3. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
4. Chowdhary, K.: Natural language processing. Fundamentals of artificial intelligence pp. 603–649 (2020)
5. Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., Stoyanov, V.: Unsupervised cross-lingual representation learning at scale. arXiv preprint arXiv:1911.02116 (2019)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
7. Dieng, A.B., Wang, C., Gao, J., Paisley, J.: Topicrnn: A recurrent neural network with long-range semantic dependency. arXiv preprint arXiv:1611.01702 (2016)
8. Fredrikson, M., Jha, S., Ristenpart, T.: Model inversion attacks that exploit confidence information and basic countermeasures. In: Proceedings of the 22nd ACM SIGSAC conference on computer and communications security. pp. 1322–1333 (2015)
9. Fredrikson, M., Lantz, E., Jha, S., Lin, S., Page, D., Ristenpart, T.: Privacy in pharmacogenetics: An {End-to-End} case study of personalized warfarin dosing. In: 23rd USENIX Security Symposium (USENIX Security 14). pp. 17–32 (2014)
10. Guzella, T.S., Caminhas, W.M.: A review of machine learning approaches to spam filtering. Expert Systems with Applications **36**(7), 10206–10222 (2009)
11. Hochreiter, S.: The vanishing gradient problem during learning recurrent neural nets and problem solutions. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **6**(02), 107–116 (1998)
12. Hunt, T., Song, C., Shokri, R., Shmatikov, V., Witchel, E.: Chiron: Privacy-preserving machine learning as a service. arXiv preprint arXiv:1803.05961 (2018)
13. Kirkpatrick, M.: Facebook's zuckerberg says the age of privacy is over. New York Times (2010), https://archive.nytimes.com/www.nytimes.com/external/readwriteweb/2010/01/10/10readwriteweb-facebooks-zuckerberg-says-the-age-of-privac-82963.html
14. Lewis, D.D., Jones, K.S.: Natural language processing for information retrieval. Communications of the ACM **39**(1), 92–101 (1996)
15. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
16. Liu, Z., Lin, W., Shi, Y., Zhao, J.: A robustly optimized bert pre-training approach with post-training. In: China National Conference on Chinese Computational Linguistics. pp. 471–484. Springer (2021)
17. Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. pp. 142–150. Association for Computational Linguistics, Portland, Oregon, USA (June 2011), http://www.aclweb.org/anthology/P11-1015

18. Nakamura, Y., Hanaoka, S., Nomura, Y., Hayashi, N., Abe, O., Yada, S., Wakamiya, S., Aramaki, E.: Kart: Privacy leakage framework of language models pre-trained with clinical records. arXiv preprint arXiv:2101.00036 (2020)
19. Pan, X., Zhang, M., Ji, S., Yang, M.: Privacy risks of general-purpose language models. In: 2020 IEEE Symposium on Security and Privacy (SP). pp. 1314–1331. IEEE (2020)
20. Price, W.N., Cohen, I.G.: Privacy in the age of medical big data. Nature medicine **25**(1), 37–43 (2019)
21. Sarker, I.H.: Machine learning: Algorithms, real-world applications and research directions. SN Computer Science **2**(3), 1–21 (2021)
22. Severyn, A., Moschitti, A.: Twitter sentiment analysis with deep convolutional neural networks. In: Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval. pp. 959–962 (2015)
23. Song, C., Raghunathan, A.: Information leakage in embedding models. In: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. pp. 377–390 (2020)
24. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. Advances in neural information processing systems **27** (2014)
25. de Vries, W., van Cranenburgh, A., Bisazza, A., Caselli, T., van Noord, G., Nissim, M.: Bertje: A dutch bert model. arXiv preprint arXiv:1912.09582 (2019)