AnonyMate: A Toolkit for Anonymizing Unstructured Chat Data

**Allison Adams, Eric Aili, Daniel Aioanei, Rebecca Jonsson, Lina Mickelsson,
Dagmar Mikmekova, Fred Roberts, Javier Fernandez Valencia, Roger Wechsler**
Artificial Solutions
Stureplan 15, Stockholm 111 45
r&d@artificial-solutions.com

Abstract

Most existing research on the automatic anonymization of text data has been limited to the de-identification of medical records. This is beginning to change following the passage of GDPR privacy laws, which have made the task of automatic text anonymization more relevant than ever. We present our privacy protection toolkit, AnonyMate, which is built to anonymize both personal identifying information (PII) as well as corporate identifying information (CII) in human-computer dialogue text data.

1 Introduction

Many NLP systems require vast amounts of text data to develop. This poses a considerable challenge to companies who want to prioritize the data integrity and privacy of their clients while building state of the art tools. The General Data Protection Regulation (GDPR) [1] sets restrictions on the usage and storage of personal identifying information (PII), which is often present in human-computer dialog data. As such, steps to remove sensitive information through anonymization are essential if the data are to be collected and stored for research and development purposes. To address this need, we developed our anonymization tool, AnonyMate, with two main objectives in mind:

- To ensure that historical data stored for R&D purposes do not contain any PII data.
- To enable our platform to produce anonymized data.

In light of these objectives, our goal was to build a tool that can identify and classify types of PII data and apply different anonymization and pseudonymization strategies on the detected PII types. We further sought to detect and annotate named entities beyond the scope of anonymization purposes.

The development of this system encompassed a diverse range of tasks including: establishing a tag set of PII and named entity types with guidelines for annotation, the creation of an annotation tool, a large-scale annotation effort in multiple languages, and the testing and implementation of Named Entity Recognition (NER) and language identification systems. The resulting anonymization pipeline comprises five modules: a pre-processing step, a language detector, an NER component, coreference resolution and, finally, an anonymization step, in which identified entities are removed or replaced. In this paper we present an overview of this project and our anonymization pipeline architecture.

2 Tag set and annotation

2.1 Tag set

In the first phase of this project, we established a set of entity types we wanted our system to be able to identify. As our data, sampled from historical chat logs, belong to a diverse set of domains, we identified 24 named entity types we expected to be present in our data. We classified them into three categories:

i. *Personal Identifying Information* (PII), or named entities that could link the data to a specific individual.

ii. *Corporate Identifying Information* (CII), or named entities that could link the data to a specific organization or client.

iii. *Other*, which contain entities we do not expect to anonymize, but nonetheless want to identify in our data.

[1] https://eur-lex.europa.eu/eli/reg/2016/679/oj

Proceedings of the Workshop on NLP and Pseudonymisation, pages 1–7,
Turku, Finland, 30 September 2019. © 2019 Linköping University Electronic Press

PII	**CII**	**Other**
Person	Organization	Nationality
Address	Product	Geographical
Zip Code	Facility	Event
Location	URL	Work of Art
Email		Language
UID		Unit
IP Address		Misc
(Date)		Med/Chem
		Sports Team
		Known Group
		Known Figure
		Fictional Figure
		Date

Table 1: Categorization of entity types in our tag set

Table 1 lists these entity types and their respective groupings. The first group, PII, comprises entities relating to an identifiable person. This category includes person names, addresses (including e-mail and IP addresses), zip codes, locations, unique identifiers (UID), which includes entities such as phone numbers or social security numbers, and in some cases birth dates. We further aimed to protect not only the privacy of individuals present in our data, but that of our corporate clients as well. The list of entity types pertaining to CII includes organizations, products, facilities and URLs. Finally, we established a list of named entities we expect to occur frequently in our data that fall outside the scope of this anonymization task. This list includes named entities useful to identify within our platform, for example for slot-filling purposes, such as nationalities, languages, units (when in the context of an amount, e.g. *5 kilometers*), medical/chemical entities, known figures, etc. We also reserved a placeholder *Miscellaneous* tag to annotate things that are clearly named entities but that do not fit in any other category, such as *What is the 50th digit of* ***Pi*** or *When did the* ***Titanic*** *sink?*.

2.2 Data selection and pre-annotation

We expected named entities to be somewhat sparsely represented in our data and, as such, to speed up the annotation process, we sought to develop a method of pre-selecting sentences for our training set that had a higher likelihood of containing a named entity. Lingren et al., 2013 have demonstrated dictionary-based annotation methods to save time on NER annotation tasks without introducing bias to the annotation process. Following these findings, we used our in-house lexical resources to develop a rule-based and dictionary-based method for identifying inputs likely to contain an entity. This system further acts as a simplistic NER tagger that pre-annotates the data.

2.3 Annotation guidelines and training

More than 15 annotators contributed to the development of our annotated NER data set, working in 6 languages (English, German, Swedish, Spanish, Italian and French). To coordinate this annotation effort we established a set of guidelines for each language, designed to be as synchronized as possible across all development languages. As a part of these guidelines, we instructed annotators to:

- Tag according to context, selecting the most obvious and probable meaning or tag in cases of ambiguous inputs (e.g. *I paid with my visa_PRODUCT* vs. *Visa_ORGANIZATION is a credit card company.*).

- Follow word boundaries in the case of compounds. This means that in English, for example, we only annotate the named entity part of the compound in *visa_PRODUCT card_X* while for Swedish *visakort_PRODUCT*, the whole compound is annotated.

- Generally, determiners are not to be included in the scope of an entity. Only annotate determiners (or other function words) if they are part of the official name of an entity, e.g. *I read the_PRODUCT times_PRODUCT*.

We further established recommendations for tags such as *Work of Art* or *Known Figure*, which require the annotator to make a subjective judgment. These guidelines include rules of thumb for what or who does or does not constitute a work of art or a known figure, where to draw the distinction between a geographical entity or a location, etc. As we used IOB encoding (Ramshaw and Marcus, 1999), a text chunking format used to denote the scope of entity chunks, to annotate our data set, we also provided instructions to our annotators on determining the start and end of an entity.

	Training Set			Test Set		
	Entities	Tokens	Sentences	Entities	Tokens	Sentences
English	62231	586637	61081	5217	51078	5097
French	33075	382099	28033	5889	60646	4914
German	73052	570527	78261	4083	30768	3949
Italian	42494	404078	39609	5565	50730	4589
Spanish	35583	357045	34684	4495	34451	4437
Swedish	53218	524703	60830	2862	24006	2763

Table 2: Training and test data set size by language

After establishing our tag set and annotation guidelines, we held training sessions with our annotators, who we in turn tasked with annotating a 300 sentence subset of the training data. We then collectively discussed the sentences for which our annotators had produced different annotations, revisiting problematic tags and reviewing the guidelines. As an additional step to improve inter-annotator agreement, we encouraged annotators to work collaboratively to reach joint decisions about difficult or ambiguous tags.

To evaluate inter-annotator agreement, we measured agreement separately for every pair of annotators on the 300 double-annotated sentences of the training set using Cohen's kappa (Cohen, 1960) and report the average score. These results are shown in Table 3.

	Average κ	Annotators
English	.89	9
French	.89	3
German	.84	6
Italian	.89	2
Spanish	.75	4
Swedish	.90	5

Table 3: Average Cohen's kappa for inter-annotator agreement

2.4 Annotation tool

In addition to receiving training in our annotation guidelines, our annotators were also instructed on how to use our web-based tool developed in-house to facilitate the process of annotating written language data. In the annotation tool user interface, the annotator chooses the appropriate label for each word in a sentence from a drop-down menu. The tool also allows the annotator to navigate through examples, giving them the option to skip tricky examples and revisit them later.

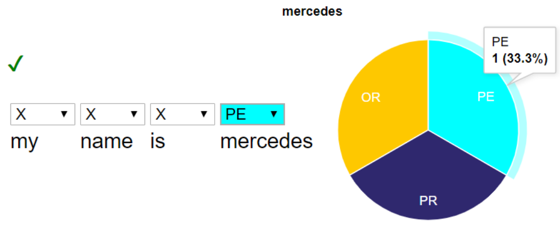

Figure 1: Annotation tool user interface

In order to ensure consistent annotation, the tool displays statistics for how a given word has been annotated previously. For instance, in the hypothetical example shown in Figure 1, the annotator can see that the ambiguous token, *Mercedes*, has been marked as a product, organization, and as a person. A regex search function then allows the user to review previous examples to see the context in which these tags were assigned.

2.5 Data sets composition

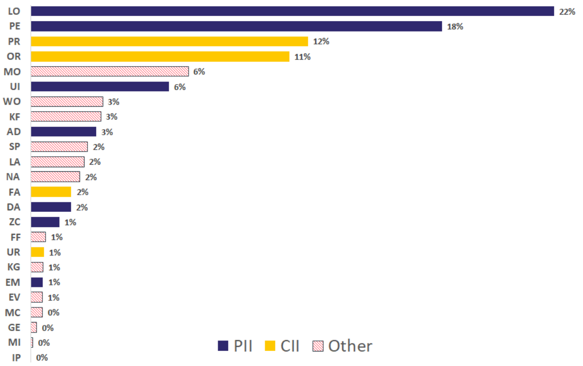

Figure 2: Distribution of named entity tags in the English training set

Table 2 features the training and test set sizes for the six languages we developed. The table lists the number of entities, tokens, and sentences that each data set contains. Our training data sets range in size from 28,033 sentences for French, to 78,261 for German. We did not necessarily expect a correlation between training data set size and NER model performance, as our larger data sets tend to contain a broader range of domains, which we expected to make them more difficult to predict.

Our English training data set contains 61,081 sentences, 585,773 tokens and 62,231 annotated entities. Figure 2 shows the distribution of named entity tag types in the English training data set. We generally observed very similar distribution pattern across all languages we developed. We opted to maintain the natural distribution of entity types in our data set, rather than artificially inflate the training set for underrepresented tag types. As Figure 2 shows, PII and CII tags occur most frequently in the data, with the exception of URLs, IP addresses and E-mail addresses. Given the predictability of these entity forms, however, we did not expect their lack of frequency in the training data to be problematic.

3 Named entity recognition for anonymization

Named entity recognition (NER), the identification of named entities in unstructured text, is a standard component of anonymization and de-identification systems. Most prior research in automatic text anonymization has focused on the de-identification of medical records, and has employed either rule-based (Ruch et al., 2000; Neamatullah et al., 2008) or machine learning (Guo et al., 2006; Yang and Garibaldi, 2015) NER techniques. For the purposes of our system, we opted for the latter and explored two different NER system architectures: one based on conditional random fields (CRFs) and the other using deep-learning techniques based on the model proposed by Lample et al., 2016, which is a BiLSTM with a CRF decoding layer. We developed the CRF model using CRFSuite (Okazaki, 2007). The neural network model was implemented in Tensorflow (Abadi et al., 2015).

In addition to using word, basic prefix and suffix, as well as regex features to help detect e-mail addresses and series of digits, one CRFSuite model makes use of embeddings clusters, which we derived by performing K-means clustering on word embeddings, which we trained on our own in-house data using Word2Vec (Mikolov et al., 2013). In doing so, our aim was to group together words which are distributionally similar in order to imbue our model with some degree of semantic understanding, while maintaining the model size small relative to using the full emebeddings model.

3.1 NER performance

SYSTEM TYPE	F1
Baseline (unamb)	45.0
Baseline (freq)	57.5
CRFSuite	74.1
CRFSuite + embeddings clusters	76.0
BiLSTM + CRF decoding	**79.2**

Table 4: NER system performance for English

Table 4 shows the results of an evaluation of our English NER models on a separate test set. We performed our evaluation following the same methods used in the CoNLL-2003 shared task on named entity recognition (Sang and De Meulder, 2003). The test set contains 5,217 entities, and comprises 51,078 tokens and 5,097 sentences, making it slightly less than 10 percent the size of the training set. We evaluated our models against two baseline metrics; an unambiguous baseline (unamb), in which entities that appear in the training set with only one annotation are assigned that label in the test set, and a frequency-based baseline (freq), in which entities that appear in the training set are assigned the most frequent annotation that the entity was given in the training set. All three models we investigated performed well over these baselines, with the highest performing model being our neural network based system. We further see that the use of embeddings clusters in the CRFSuite model results in a modest improvement in F1 compared to not using the embeddings clusters. We used default parameters when training and testing these models, so it is possible that tuning could lead to further improvements over the baseline.

Figure 3 shows the F1 per named entity tag of the CRFSuite model with word embeddings clusters for English. The highest performing entity types benefit from our regex pattern matching feature, which identifies sequences of digits and special characters. Moreoever, we see F1 scores of 75% and above for all PII entity types, and 70% and above for all CII entity types. Table 5 shows averaged precision, recall and F1 for PII

LANGUAGE	PII			CII		
	P	R	F1	P	R	F1
English	86.4	82.7	84.6	87.0	73.5	79.5
French	89.4	89.4	89.3	85.3	78.0	81.0
German	92.7	89.6	91.1	86.5	65.5	73.5
Italian	88.7	86.0	87.1	87.5	73.0	77.8
Swedish	89.1	83.7	86.1	84.5	71.5	77.0
Spanish	89.1	86.9	87.7	89.3	80.5	84.5

Table 5: Average PII and CII performance: English CRFSuite with embeddings clusters

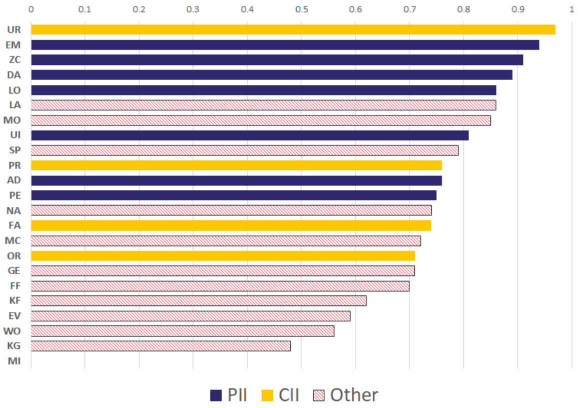

Figure 3: Performance of English CRFSuite with embeddings clusters, by named entity type

LANGUAGE	CRF	Freq.	Unamb.
English	76.0	57.5	45.0
French	84.9	75.3	64.5
German	85.4	70.8	59.1
Italian	83.8	72.9	67.7
Spanish	80.8	68.9	57.7
Swedish	76.5	62.5	52.8

Table 6: NER performance by language: CRFSuite with embeddings clusters

and CII for each language. As the table shows, both PII and CII types perform well above the average model F1. Our evaluations are carried out on the chunk level, rather than on token level, and we observe that scores are generally lower for tags likely to contain multi-token entities (e.g. personal names, addresses, facilities, organizations, etc.). A point of further investigation is to perform an error analysis on these entity types, as even partial recognition of an entity chunk is likely to be sufficient for anonymization purposes. We further observe a correlation between entity tag frequency in the data set and performance, suggesting that the performance of some tags could be improved through the addition of training data for these entity types. As IP addresses were generally lacking from our data set, we opted to remove this tag from our NER training set and use regular expressions instead of relying on NER.

Finally, Table 6 shows the performance of the CRFSuite model with embeddings clusters for each language as compared to the two baseline evaluation metrics. As the table shows, the models for all languages performed well over both baselines. We do, however, see that performance gains over the baseline are more modest for the languages for which we have less training data.

4 Language detector

Given that our anonymization pipelines are language-specific, in order to ensure we anonymize our data effectively, we developed an automatic language identification system to confirm that inputs are being sent into the correct NER pipeline. Our data are organized according to project, which are typically monolingual, however we expect a certain amount of noise in the data, and want to be sure that we do not fail to anonymize PII based on this factor.

Our language detector is currently capable of predicting 45 languages and was trained using OpenNLP's (Apache Software Foundation, 2014) language detector model (a Naïve Bayes Classifier) on a training set of 182,087 sentences. We sourced the training data from a combination of in-house project data as well as external corpora, namely, the OpenSubtitles (Tiedemann, 2016) and Europarl (Koehn, 2005) corpora. We cleaned our in-house data in the following ways:

- An initial coarse regex-based method to identify English inputs based on frequently occurring words (e.g. *Hello*, *would*, *could*, etc.).

- Analyzing a preliminary model's output on the data set using cross-validation to identify sentences incorrectly classified as false positives.

These adjustments to our training data resulted in a final F1 of 93.01% tested on separate test set of 19,828 sentences.

5 Coreference resolution

The last stage in our pipeline before anonymization handles basic coreference resolution. This system keeps track of multiple occurrences of entities on a user chat session level. For example, if a user refers to the same person name multiple times throughout a chat session, the name is anonymized to *Person 1*. If a user then mentions a second name during the course of a session, that name is then anonymized to *Person 2*. This allows us to maintain the distinction between different individuals while protecting the privacy of those discussed over the course of a full dialogue.

6 Anonymization pipeline

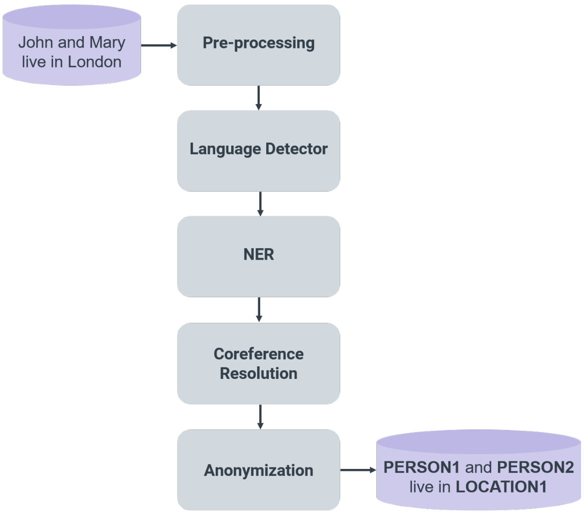

Figure 4: Anonymization pipeline architecture

Figure 4 shows the AnonyMate pipeline architecture. An input is first sent to a pre-processing module which deunicodes, removes non-printable characters, and strips HTML tags before tokenizing the input. The input is then sent to the language detector. Inputs identified as foreign are deleted from our logs rather than being sent to the NER module. Depending on the settings selected, the input can be sent either to be processed by a BiLSTM+CRF NER module or a CRF NER module. Finally, after the input has been analyzed for entities, coreference resolution is applied to the input.

The anonymization strategy applied is configurable by the user, where the user can select which entity types to anonymize. Moreover, the tool allows the option to suppress certain entity types, whereby entities are simply removed from the input (e.g. *I live in London.* → *I live in* *** .); tag entities, in which entities are replaced with their named entity tag (e.g. *I live in London.* → *I live in LOCATION.*); or substitute entities, in which a specific entity is replaced by a predetermined string (e.g. *I live in London.* → *I live in ENGLISH_CITY.*)

7 Conclusion

In this paper we have presented an overview of our anonymization toolkit, AnonyMate, and detailed the stages of the project. We have described the creation of a tag set and data set used to train and test a named entity recognition system that can be applied to the tasks of anonymization and slot-filling, as well as given an evaluation of the NER systems we developed. We further reported on the implementation of a language detection system used to filter foreign inputs that our language-specific anonymization pipeline would fail to successfully de-identify. Finally, we provided a description of the anonymization pipeline architecture, and discussed the various strategies employed to remove personal and corporate identifying information from our data. AnonyMate has given us the ability to both remove PII and CII data from our historical data, so that they can be stored for future use in research and development, as well as enabled our platform to generate anonymized data.

Acknowledgments

We would like to thank our annotators for their hard work and dedication in creating our data sets, as well as the three reviewers for their valuable comments.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. http://tensorflow.org/ TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Apache Software Foundation. 2014. http://opennlp.apache.org/ openNLP Natural Language Processing Library. Http://opennlp.apache.org/.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Yikun Guo, Robert Gaizauskas, Ian Roberts, George Demetriou, Mark Hepple, et al. 2006. Identifying personal health information using support vector machines. In *i2b2 workshop on challenges in natural language processing for clinical data*, pages 10–11. Citeseer.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Todd Lingren, Louise Deleger, Katalin Molnar, Haijun Zhai, Jareen Meinzen-Derr, Megan Kaiser, Laura Stoutenborough, Qi Li, and Imre Solti. 2013. https://doi.org/10.1136/amiajnl-2013-001837 Evaluating the impact of pre-annotation on annotation speed and potential bias: Natural language processing gold standard development for clinical named entity recognition in clinical trial announcements. *Journal of the American Medical Informatics Association : JAMIA*, 21.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Ishna Neamatullah, Margaret M Douglass, H Lehman Li-wei, Andrew Reisner, Mauricio Villarroel, William J Long, Peter Szolovits, George B Moody, Roger G Mark, and Gari D Clifford. 2008. Automated de-identification of free-text medical records. *BMC medical informatics and decision making*, 8(1):32.

Naoaki Okazaki. 2007. http://www.chokkan.org/software/crfsuite/ Crfsuite: a fast implementation of conditional random fields (crfs).

Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Patrick Ruch, Robert H Baud, Anne-Marie Rassinoux, Pierrette Bouillon, and Gilbert Robert. 2000. Medical document anonymization with a semantic lexicon. In *Proceedings of the AMIA Symposium*, page 729. American Medical Informatics Association.

Erik F Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

Jörg Tiedemann. 2016. Finding alternative translations in a large corpus of movie subtitle. In *LREC*.

Hui Yang and Jonathan M Garibaldi. 2015. Automatic detection of protected health information from clinic narratives. *Journal of biomedical informatics*, 58:S30–S38.