Language Invariant Properties in Natural Language Processing

Federico Bianchi, Debora Nozza, Dirk Hovy
Bocconi University
Via Sarfatti 25
Milan, Italy
{f.bianchi,debora.nozza,dirk.hovy}@unibocconi.it

Abstract

Meaning is context-dependent, but many properties of language (should) remain the same even if we transform the context. For example, sentiment or speaker properties should be the same in a translation and original of a text. We introduce **language invariant properties**: i.e., properties that should not change when we transform text, and how they can be used to quantitatively evaluate the robustness of transformation algorithms. Language invariant properties can be used to define novel benchmarks to evaluate text transformation methods. In our work we use translation and paraphrasing as examples, but our findings apply more broadly to any transformation. Our results indicate that many NLP transformations change properties. We additionally release a tool as a proof of concept to evaluate the invariance of transformation applications.

1 Introduction

The progress in Natural Language Processing has bloomed in recent years, with novel neural models being able to beat the score of different benchmarks. However, current evaluation benchmarks often do not look at how properties of language vary when text is transformed or influenced by a change in context. For example, the meaning of a sentence is influenced by a host of factors, among them who says it and when: “That was a sick performance” changes meaning depending on whether a 16-year-old says it at a concert or a 76-year-old after the opera.[1] However, there are several properties of language that do (or should) not change when we *transform* a text (i.e., change the surface form of it to another text, see also Section 2). If the text was written by a 25-year-old female, it should not be perceived as written by an old man after we apply a paraphrasing algorithm. The same goes for other properties, like sentiment: A positive message like “good morning!”, posted on social media, should be perceived as a positive message, even when it is translated into another language.[2] We refer to these properties that are unaffected by transformations as **Language Invariant Properties (LIPs)**. LIPs preserve the semantics and pragmatic components of language. I.e., these properties are not affected by transformations applied to the text. For example, we do not expect a summary to change the topic of a sentence.

Paraphrasing, **summarization**, **style transfer**, and **machine translation** are all NLP transformation tasks that should respect LIPs. If they do not, it is a strong indication that the system is picking up on spurious signals and needs to be recalibrated. For example, machine translation should not change speaker demographics or sentiment, and paraphrasing should not change entailment or topic.

But what happens if a transformation *does* violate invariants? Violating invariants is similar to breaking the cooperative principle (Grice, 1975): if we do it deliberately, we might want to achieve an effect. For example, Reddy and Knight (2016) showed how words can be replaced to obfuscate author gender, thereby protecting their identity. Style transfer can therefore be construed as a deliberate violation of LIPs. In most cases, though, violating a LIP will result in an unintended outcome or interpretation of the transformed text: for example, violating LIPs on sentiment will generate misunderstanding in the interpretation of messages. Any such violation might be a signal that models are not ready for production (Bianchi and Hovy, 2021).

In this paper, we suggest a novel type of evaluation benchmark based on LIPs. We release a tool as a proof of concept of how this methodology can be introduced into evaluation pipelines: we define the concept of LIPs, but also integrate

[1] Example due to Veronica Lynn.

[2] https://gu.com/technology/2017/oct/24/facebook-palestine-israel-translates-good-morning-attack-them-arrest

Proceedings of NLP Power! The First Workshop on Efficient Benchmarking in NLP, pages 84 - 92
May 26, 2022 ©2022 Association for Computational Linguistics

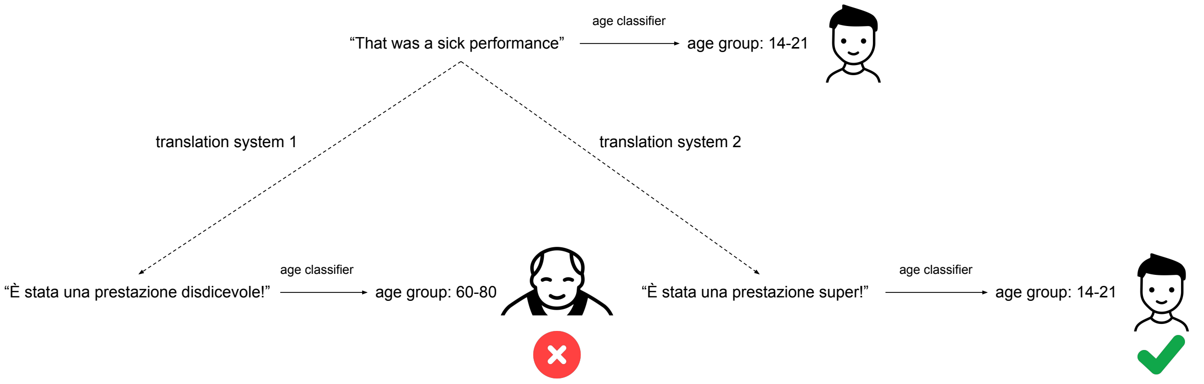


Figure 1: Author age is a Language Invariant Property (LIP). Translation system 1 fails to account for this and provides a translation that can give the wrong interpretation to the sentence. Translation system 2 is instead providing a more correct interpretation.

insights from Hovy et al. (2020), defining an initial benchmark to study LIPs in two of the most well-known transformation tasks: machine translation and paraphrasing. We apply those principles more broadly to transformations in NLP as a whole.

Contributions. We introduce LIPs: properties of language that should not change during a transformation. Our contribution also focuses on the proposal of an evaluation methodology for LIPs and the release of a Python application that can be used to test how well systems can preserve LIPs.[3] We believe that this contribution can help the community to work on benchmarking and understanding how properties change when text is transformed.

2 Language Invariant Properties

To use the concept of LIPs, we first need to make clear what we mean by it. We formally define LIPs and transformations below.

Assume the existence of a set S of all the possible utterable sentences. Let us define A and B as subsets of S. These can be in the same or different languages. Now, let's define a mapping function

$$t : A \rightarrow B$$

i.e., $t(\cdot)$ is a **transformation** that changes the surface form of the text A into B.

A *language property* p is a function that maps elements of S to a set P of property values. p is **invariant** if and only if

$$\forall a \in A \;\; p(a) = p(t(a)) = p(b)$$

where $b \in B$, and $t(a) = b$. I.e., if applying $p(\cdot)$ to both an utterance and its transformation still maps to the same property. We do not provide an exhaustive list of these properties, but suggest to include at least **meaning**, **topic**, **sentiment**, **speaker demographics**, and **logical entailment**.

LIPs are thus based on the concept of text transformations. Machine translation (MT) is a salient example of a transformation and a prime example of a task for which LIPs are important. MT can be viewed as a transformation between two languages where the main fundamental LIP that should not be broken is meaning.

However, LIPs are not restricted to MT but have broader applicability, e.g., in style transfer. In that case, though, some context has to be defined. When applying a *formal* to *polite* transfer, this function is by definition *not* invariant anymore. Nonetheless, many other properties should not be influenced by this transformation. Finally, for paraphrasing, we have only one language, but we have the additional constraint that $t(a) \neq a$. For summarization, the constraint instead is that $len(t(a)) < len(a)$.

LIPs are also what make some tasks in language more difficult than others: for example, data augmentation (Feng et al., 2021) cannot be as easily implemented in text data as in image processing, since even subtle changes to a sentence can affect meaning and style. Changing the slant or skew of a photo will still show the same object, but for example word replacement easily breaks LIPs, since the final meaning of the final sentence and the perceived characteristics can differ. Even replacing a

[3] https://github.com/MilaNLProc/language-invariant-properties

word with one that is similar can affect LIPs. For example, consider machine translation with a parallel corpus: "the dogs are running" can be paired with the translation "I cani stanno correndo" in Italian. If we were to do augmentation, replacing *dogs* with its hyperonym *animals* does not corrupt the overall meaning, as the new English sentence still entails all that is entailed by the old one. However, the Italian example is no longer a correct translation of the new sentence, since *cani* is not the word for animals.

LIPs are also part of the communication between speakers. The information encoded in a sentence uttered by one speaker contains LIPs that are important for efficient communication, as misunderstanding a positive comment as a negative one can create issues between communication partners.

Note that we are not interested in evaluating the *quality* of the transformation (e.g., the translation or paraphrase). There are many different metrics and evaluation benchmarks for that (BLEU, ROUGE, BERTscore etc.: Papineni et al., 2002; Lin, 2004; Zhang et al., 2020b). Our analysis concerns another aspect of communication for which we wish to propose an initial benchmark.

3 Related Work

There have been different works in NLP that have investigated issues arising from language technology (Hovy and Spruit, 2016; Blodgett et al., 2020; Bianchi and Hovy, 2021; Bolukbasi et al., 2016; Gonen and Goldberg, 2019; Lauscher et al., 2020; Bianchi et al., 2021a; Dev et al., 2020; Sheng et al., 2019; Nozza et al., 2021, 2022). In our paper, we focus on issues that can arise from the usage of text transformation algorithms (for example, we will see examples of gender bias in transformation, inspired by (Hovy et al., 2020), in Section 5) and we describe a method that can allow us to analyze them.

The idea that drives LIPs have spawned across different work in the NLP literature; For example, Poncelas et al. (2020) discuss the effect that machine translation can have on sentiment classifiers. At the same time, ideas of conserving meaning during style transfer are also presented in the work by Hu et al. (2020). We propose LIPs as a term to give a unified view on the problem of preserving these properties during transformation.

LIPs share some notions with the checklist by Ribeiro et al. (2020) and the adversarial reliability checks by Tan et al. (2021). However, LIPs evaluate how well fundamental properties of discourse are preserved in a transformation, the checklist is made to guide users in a fine-grained analysis of the model performance to better understand bugs in the applications with the use of templates. As we will show later, LIPs can be quickly tested to any new annotated dataset. Some of the checklist's tests, like *Replace neutral words with other neutral words*, can be seen as LIPs. The general idea of adversarial attacks, meanwhile, also requires LIPs to hold in order to work. Nonetheless, we think the frameworks are complementary.

4 Benchmarking Transformation Invariance

For ease of reading, we will use translation as an example of a transformation in the following. However, the concept can be applied to any of the transformations we mentioned above.

We start with a set of original texts A to translate into a set of texts B.[4] We thus need a translation model t from the source language of A to a target language of B. To test the transformation wrt a LIP, A should be annotated with that language property of interest, this is our ground truth and we are going to refer to this as $\hat{p}(A)$. We also need a classifier for the LIP of interest, which serves as language property function p. For example, a LIP classifier could be a gender classifier that, given an input text, returns the inferred gender of the speaker. Here, we need one cross-lingual classifier, or two classifiers, one in the source and one in the target language.[5]

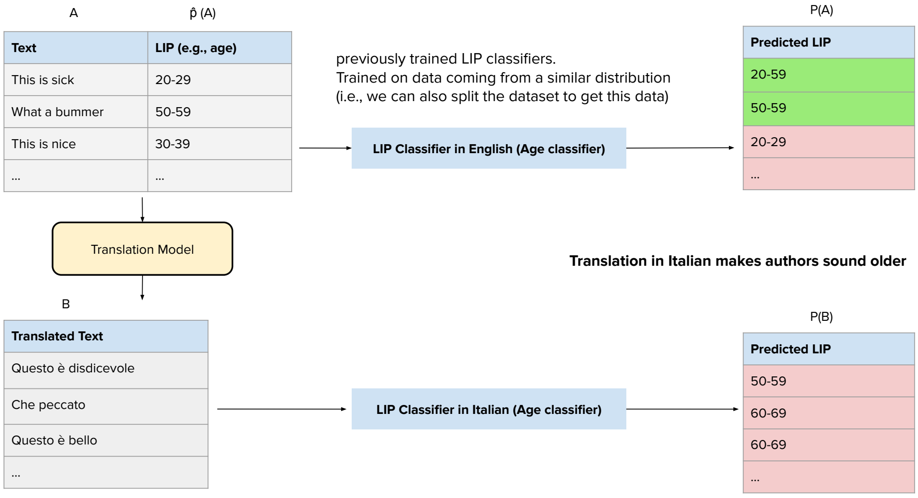
Once we apply the translation, we can use the LIP classifier on the original data A and the new set of translated data B obtaining respectively, $p(A)$ and $p(B)$.

We can then compare the difference between the distribution of the LIP in the original data and either prediction. I.e., we compare the differences in distribution of $\hat{p}(A) - p(A)$ to $\hat{p}(A) - p(B)$ to understand the effect of the transformations. We show a visual explanation on how to benchmark LIPs in Figure 2.

Note that we are *not* interested in the actual performance of the classifiers, but in the difference in performance on the two datasets. We observe two possible phenomena (as in Hovy et al. (2020)):

[4] We slightly abuse of notation here and interpret A has the set of original texts instead of the set of the possible utterances.

[5] For all other transformations, which stay in the same language, we only need one classifier. (Paraphrasing or summarization can be viewed as a transformation from English to English).


A
p̂ (A)
previously trained LIP classifiers.
Trained on data coming from a similar distribution
(i.e., we can also split the dataset to get this data)
LIP Classifier in English (Age classifier)
P(A)
Translation Model
Translation in Italian makes authors sound older
B
LIP Classifier in Italian (Age classifier)
P(B)

Text	LIP (e.g., age)
This is sick	20-29
What a bummer	50-59
This is nice	30-39
...	...

Predicted LIP
20-59
50-59
20-29
...

Translated Text
Questo è disdicevole
Che peccato
Questo è bello
...

Predicted LIP
50-59
60-69
60-69
...

Figure 2: A visual explanation on how to benchmark LIPs.

1) If there is a *classifier bias*, both the predictions based on the original language and the predictions based on the translations should be skewed in the **same** direction with respect to the distribution in A. E.g., for gender classification, both classifiers predict a higher rate of male authors in the original and the translated text. 2) Instead, if there is a *transformation bias*, then the distribution of the translated predictions should be skewed in a different direction than the one based on the original language. E.g., the gender distribution in the original language should be less skewed than the gender ratio in the translation.

Note that we assume that the LIP classifiers used for the source and one in the target language have similar biases; if this were not true and the classifiers had different biases phenomena 1) could be caused both by the bias in translations or bias in the models. This mostly depends on the quality of the classifiers, that has to be assessed before the evaluation of the LIPs.

4.1 Datasets

Here, we evaluate machine translation and paraphrasing as transformation tasks. Our first release of this benchmark tool contains the datasets from Hovy et al. (2020), annotated with gender[6] and age categories, and the SemEval dataset from Mohammad et al. (2018) annotated with emotion recognition. Moreover, we include the English dataset from HatEval (Basile et al., 2019) containing tweets for hate speech detection. These datasets come with training and test splits and we use the training data to train the LIP classifiers.

Nonetheless, our benchmark can be easily extended with new datasets encoding other LIPs.

4.2 TrustPilot

We use a subset of the dataset by Hovy et al. (2015). It contains TrustPilot reviews in English, Italian, German, French, and Dutch with demographic information about the user's age and gender. Training data for the different languages consists of 5k samples (balanced for gender) and can be used to build the LIP classifiers. The dataset can be used to evaluate the LIPs AUTHOR-GENDER and AUTHOR-AGE.

4.3 HatEval

We use the English tweet data from HatEval (Basile et al., 2019). We take the training and test set (3k examples). Each tweet comes with a value that indicates if the tweet contains hate speech. The dataset can be used to evaluate the LIP HATEFULNESS.

4.4 Affects in Tweets (AiT)

We use the Affect in Tweets dataset (AiT) (Mohammad et al., 2018), which contains tweets annotated with emotions. We reduce the number of possible classes by only keeping emotions in the set {*fear, joy, anger, sadness*} to allow for future comparisons with other datasets. We then map *joy* to *positive* and the other emotions to *negative* for deriving the sentiment following Bianchi et al. (2021b, 2022). The data we collected comes in English (train: 4,257, test: 2,149) and Spanish (train:

[6] The dataset comes with binary gender, but this is not an indication of our views or the capabilities of the tool.

Method	L1	L2	$KL_{A,p(A)}$	$KL_{B,p(B)}$	Dist $\hat{p}(A)$	Dist $p(A)$	Dist $p(B)$
SE	IT	EN	0.004	0.034	M: 0.52, F: 0.48	M: 0.56, F: 0.44	M: **0.64**, F: 0.36
TF	IT	EN	0.000	0.034	M: 0.52, F: 0.48	M: 0.53, F: 0.47	M: **0.64**, F: 0.36
SE	DE	EN	0.000	0.030	M: 0.50, F: 0.50	M: 0.49, F: 0.51	M: **0.61**, F: 0.39
TF	DE	EN	0.001	0.022	M: 0.50, F: 0.50	M: 0.52, F: 0.48	M: **0.60**, F: 0.40

Table 1: Results on TrustPilot dataset translating IT/DE–EN. TF = logistic regression classifier with TF-IDF (TF), SE = (cross-lingual) embedding model. Translation breaks the LIP AUTHOR-GENDER

2,366, test: 1,908). The dataset can be used to evaluate the LIP SENTIMENT.

4.5 Methods

Classifiers As default classifier we use L2-regularized Logistic Regression models over 2-6 TF-IDF character-grams (Hovy et al., 2020). Due to the great recent results of pre-trained language models (Nozza et al., 2020), we also use SBERT (Reimers and Gurevych, 2019) to generate sentence embeddings and use these representations as input to a logistic regression (L2 regularization and balance weights). The two classification methods are referred to as TF (TF-IDF) and SE (Sentence Embeddings). Our framework supports the use of any classifiers. The advantage of this setup is that it is generally fast to set up, but interested user can also use more complex transformer models. The replicability details appear in the Appendix.

Scoring Standard metrics for classification evaluation can be used to assess how much LIPs are preserved during a transformation. Following Hovy et al. (2020) we use the KL divergence to compute the distance - in terms of the distribution divergence - between the two predicted distributions. The benchmark also outputs the X^2 test to assess if there is a significant difference in the predicted distributions. It is also possible to look at the plots of the distribution to understand the effects of the transformations (see following examples in Figures 3, 4 and 5).

5 Evaluation

We evaluate four tasks, i.e., combinations of transformations and LIPs; the combination is determined by the availability of the particular property in the respective dataset.

5.1 TrustPilot Translation - LIP: AUTHOR-GENDER

We use the TrustPilot dataset to study the author-gender LIP during translation. We use the Google translated documents provided by the authors. We are essentially recomputing the results that appear in the work by Hovy et al. (2020). As shown in Table 1, our experiments with both TF and SE confirm the one in the paper: it is easy to see that the translations from both Italian and German into English are more likely to be predicted as male (this can be seen by the change in the distribution), breaking the LIP AUTHOR-GENDER.

5.2 AiT Translation - LIP: SENTIMENT

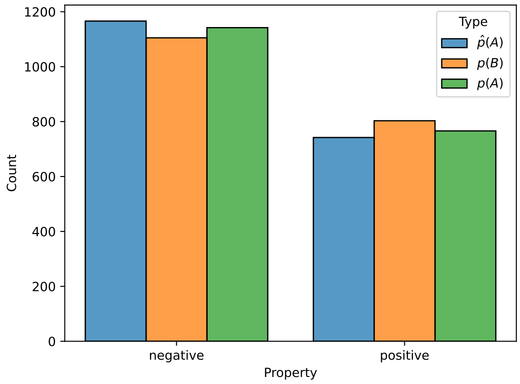


Figure 3: Translation ES–EN on AiT sentiment analysis. Translation respects the LIP SENTIMENT

We use the AiT dataset to test the sentiment LIP during translation. We translate the tweets from Spanish to English using DeepL. We use SE as our embedding method. As shown in Figure 3, SENTIMENT is a LIP that seems to be easily kept during translations. This is expected, as sentiment is a fundamental part of the meaning of a sentence and has to be translated accordingly.

5.3 TrustPilot Paraphrasing - LIP: AUTHOR-GENDER

When we apply paraphrasing on the TrustPilot data, the SE classifier on the transformed data predicts more samples as male (see Figure 4 that plots the distribution). $KL_{A,p(B)} = 0.018$, difference sig-

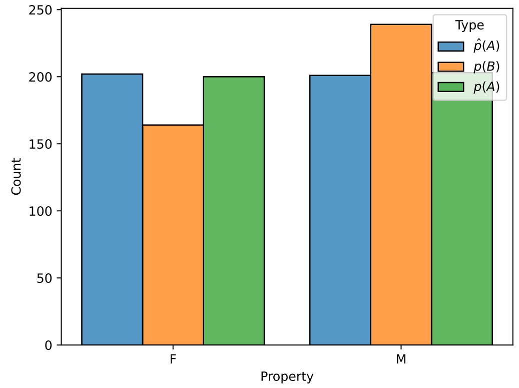


Figure 4: Paraphrasing on TrustPilot English data. Paraphrasing breaks the LIP AUTHOR-GENDER

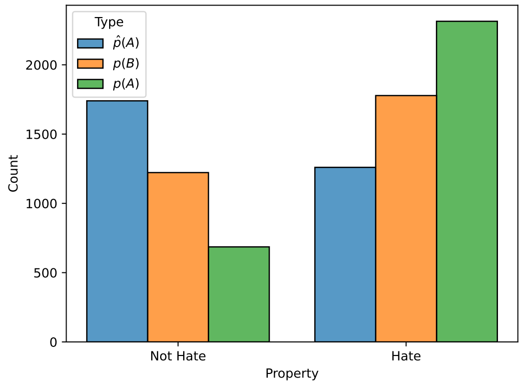


Figure 5: Paraphrasing on HatEval English data. Paraphrasing breaks the LIP HATEFULNESS

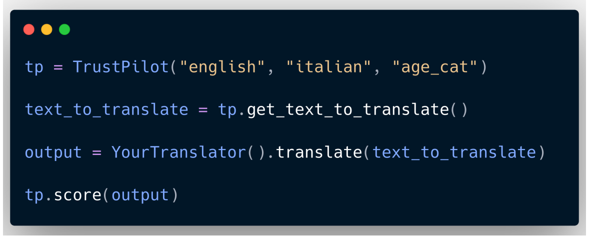
nificant for X^2 with $p < 0.01$, resulting in a break of the LIP HATEFULNESS.

5.4 HatEval Paraphrasing - LIP: HATEFULNESS

We use the HatEval data to study the hatefulness LIP after paraphrasing. We use SE as our embedding method. Figure 5 shows that the SE classifier predicted a high amount of hateful tweets in $p(A)$ (a problem due to the differences between the training and the test in HatEval (Basile et al., 2019; Nozza, 2021)), this number is drastically reduced in $p(B)$, suggesting that paraphrasing reduces hatefulness, breaking the LIP. As an example of paraphrased text, *Savage Indians living up to their reputation* was transformed to *Indians are living up to their reputation*. While the message still internalizes bias, removing the term *Savage* has reduced its strength. It is important to remark that we are not currently evaluating the *quality* of the transformation—that is another task. The results we obtain are in part due to the paraphrasing tool we used,[7] but they still indicate a limit in the model capabilities.

6 Benchmark Tool

We release an extensible benchmark tool[8] that can be used to quickly assess a model's capability to handle LIPs. The benchmark has been designed to provide a high-level API that can be integrated into any transformation pipeline. Users can access the dataset text, transform, and score it (see Figure 6). Thus, this pipeline should be very easy to use. The tool allows the users to run the experiments multiple time to also understand the variations that depends on the model themselves.



```
tp = TrustPilot("english", "italian", "age_cat")

text_to_translate = tp.get_text_to_translate()

output = YourTranslator().translate(text_to_translate)

tp.score(output)
```

Figure 6: The benchmark has been designed to provide a high-level API that can be integrated in any transformation pipeline. Users can access the dataset text, transform, and score it.

We hope this benchmark tool can be of help, even as an initial prototype, in designing evaluation pipelines meant at studying how LIPs are preserved in text.

7 Conclusion

This paper introduces the concept of Language Invariant Properties, properties in language that should not change during transformations. We also describe a possible evaluation pipeline for LIPs showcasing that some of the language transformation technologies we use suffer from limitations and that they cannot often preserve important LIPs.

We believe that the study of LIPs can improve the performance of different NLP tasks and to provide better support in this direction we release a benchmark that can help researchers and practitioners understand how well their models handle LIPs.

[7] https://huggingface.co/tuner007/pegasus_paraphrase

[8] This will be a link to a GitHub Repo

8 Limitations

The tool we implemented comes with some limitations. We cannot completely remove the learned bias in the classifiers and we always assume that when there are two classifiers, these two perform reliably well on both languages so that we can compare the output.

To reduce one of the possible sources of bias, the classifiers are currently trained with data coming from a similar distribution to the one used at test time, ideally from the same collection.

Ethical Considerations

We are aware that our work assumes that it is easy to classify text in different languages even when considering cultural differences and we do not aim to ignore this. We know that cultural differences can make it more difficult to preserve LIPs (Hovy and Yang, 2021): it might not be possible to effectively translate a positive message into a language that does not share the same appreciation/valence for the same things. However, we also believe this is a more general limitation of machine translation. The speaker's intentions are to keep the message consistent - in terms of LIPs - even when translated.

Acknowledgments

We thank Giovanni Cassani and Amanda Curry for the comments on an early draft of this work. This project has partially received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 949944, INTEGRATOR), and by Fondazione Cariplo (grant No. 2020-4288, MONICA). The authors are members of the Data and Marketing Insights Unit of the Bocconi Institute for Data Science and Analysis.

References

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Federico Bianchi and Dirk Hovy. 2021. On the gap between adoption and understanding in NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3895–3901, Online. Association for Computational Linguistics.

Federico Bianchi, Marco Marelli, Paolo Nicoli, and Matteo Palmonari. 2021a. SWEAT: Scoring polarization of topics across different corpora. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10065–10072, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Federico Bianchi, Debora Nozza, and Dirk Hovy. 2021b. FEEL-IT: Emotion and sentiment classification for the Italian language. In *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 76–83, Online. Association for Computational Linguistics.

Federico Bianchi, Debora Nozza, and Dirk Hovy. 2022. XLM-EMO: Multilingual Emotion Prediction in Social Media Text. In *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*. Association for Computational Linguistics.

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online. Association for Computational Linguistics.

Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam Tauman Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 4349–4357.

Sunipa Dev, Tao Li, Jeff M. Phillips, and Vivek Srikumar. 2020. On measuring and mitigating biased inferences of word embeddings. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7659–7666. AAAI Press.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 968–988, Online. Association for Computational Linguistics.

Hila Gonen and Yoav Goldberg. 2019. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. In *Proceedings of the 2019 Workshop on Widening NLP*, pages 60–63, Florence, Italy. Association for Computational Linguistics.

Paul Grice. 1975. Logic and conversation. In *Syntax and semantics. 3: Speech acts*, pages 41–58. New York: Academic Press.

Dirk Hovy, Federico Bianchi, and Tommaso Fornaciari. 2020. “you sound just like your father” commercial machine translation systems include stylistic biases. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1686–1690, Online. Association for Computational Linguistics.

Dirk Hovy, Anders Johannsen, and Anders Søgaard. 2015. User review sites as a resource for large-scale sociolinguistic studies. In *Proceedings of the 24th International Conference on World Wide Web*, WWW ’15, page 452–461. International World Wide Web Conferences Steering Committee.

Dirk Hovy and Shannon L. Spruit. 2016. The social impact of natural language processing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 591–598, Berlin, Germany. Association for Computational Linguistics.

Dirk Hovy and Diyi Yang. 2021. The importance of modeling social factors of language: Theory and practice. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 588–602, Online. Association for Computational Linguistics.

Zhiqiang Hu, Roy Ka-Wei Lee, Charu C Aggarwal, and Aston Zhang. 2020. Text style transfer: A review and experimental evaluation. *arXiv preprint arXiv:2010.12742*.

Anne Lauscher, Goran Glavaš, Simone Paolo Ponzetto, and Ivan Vulić. 2020. A general framework for implicit and explicit debiasing of distributional word vector spaces. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8131–8138.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. SemEval-2018 task 1: Affect in tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 1–17, New Orleans, Louisiana. Association for Computational Linguistics.

Debora Nozza. 2021. Exposing the limits of zero-shot cross-lingual hate speech detection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 907–914, Online. Association for Computational Linguistics.

Debora Nozza, Federico Bianchi, and Dirk Hovy. 2020. What the [MASK]? Making Sense of Language-Specific BERT Models. *arXiv preprint arXiv:2003.02912*.

Debora Nozza, Federico Bianchi, and Dirk Hovy. 2021. ”HONEST: Measuring hurtful sentence completion in language models”. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2398–2406, Online. Association for Computational Linguistics.

Debora Nozza, Federico Bianchi, Anne Lauscher, and Dirk Hovy. 2022. Measuring Harmful Sentence Completion in Language Models for LGBTQIA+ Individuals. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Alberto Poncelas, Pintu Lohar, James Hadley, and Andy Way. 2020. The impact of indirect machine translation on sentiment classification. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 78–88, Virtual. Association for Machine Translation in the Americas.

Sravana Reddy and Kevin Knight. 2016. Obfuscating gender in social media writing. In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 17–26, Austin, Texas. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2019. The woman worked as a babysitter: On biases in language generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the*

9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 3407–3412, Hong Kong, China. Association for Computational Linguistics.

Samson Tan, Shafiq Joty, Kathy Baxter, Araz Taeihagh, Gregory A. Bennett, and Min-Yen Kan. 2021. Reliability testing for natural language processing systems. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4153–4169, Online. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020a. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

A Logistic Regression Setup

We use a 5 fold cross-validation on the training data to select the best parameters of the logisitic regression. Class weights are balanced and we use L2 Regularization. We search the best C value in [5.0, 2.0, 1.0, 0.5, 0.1]. The solver used is *saga*.

When using TF-IDF we use the following parameters: ngram range=(2, 6), sublinear tf=True, min df=0.001, max df=0.8.

Nevertheless, the tool we will share will contain all the parameters (the tool is versioned, so it is easy to track the changes and check which parameters have been used to run the experiments).

B Models Used

B.1 TrustPilot Paraphrase

We use the same classifier for the original and the transformed text. We generate the representations with SBERT. The model used is *paraphrase-distilroberta-base-v2*.[9]

As paraphrase model, we use a fine-tuned Pegasus (Zhang et al., 2020a) model, pegasus paraphrase,[10] that at the time of writing is one of the most downloaded on the HuggingFace Hub.

B.2 AiT Translation

We translated the tweets using the DeepL APIs.[11] As classifiers we use the cross-lingual model for both languages, each language has its language-specific classifier. The cross-lingual sentence embedding method used is *paraphrase-multilingual-mpnet-base-v2*, from the SBERT package.

B.3 TrustPilot Translation

As translation we use the already translated sentences from the TrustPilot dataset provided by Hovy et al. (2020). We use both the TF-IDF based and the cross-lingual classifier, as shown in Table 1, each language has its own language-specific classifier. The cross-lingual sentence embedding method used is *paraphrase-multilingual-mpnet-base-v2*, from the SBERT package.

B.4 HatEval Paraphrasing

We use the same classifier for the original and the transformed text. We generate the representations with SBERT. The model used is *paraphrase-distilroberta-base-v2*. Users are replaced with @*user*, hashtags are removed.

As paraphrase model, we use a fine-tuned Pegasus (Zhang et al., 2020a) model, pegasus paraphrase, that at the time of writing is one of the most downloaded on the HuggingFace Hub.

[9] https://sbert.net

[10] https://huggingface.co/tuner007/pegasus_paraphrase

[11] https://deepl.com/