We Need to Talk About Random Splits

Anders Søgaard[1,2] **Sebastian Ebert**[1] **Jasmijn Bastings**[1] **Katja Filippova**[1]

[1]: Google Research, Berlin
[2]: University of Copenhagen

{soegaard|eberts|bastings|katjaf}@google.com

Abstract

Gorman and Bedrick (2019) argued for using random splits rather than standard splits in NLP experiments. We argue that random splits, like standard splits, lead to overly optimistic performance estimates. We can also split data in biased or adversarial ways, e.g., training on short sentences and evaluating on long ones. Biased sampling has been used in domain adaptation to simulate real-world drift; this is known as the covariate shift assumption. In NLP, however, even worst-case splits, maximizing bias, often under-estimate the error observed on new samples of in-domain data, i.e., the data that models should minimally generalize to at test time. This invalidates the covariate shift assumption. Instead of using multiple random splits, future benchmarks should ideally include multiple, independent test sets instead; if infeasible, we argue that multiple biased splits leads to more realistic performance estimates than multiple random splits.

1 Introduction

It is common practice in NLP to collect and annotate a text corpus – and split it into training, development and test data. These splits are often based on the order in which texts were published or sampled, and are referred to as 'standard splits'. Gorman and Bedrick (2019) recently showed that system ranking results based on *standard* splits differ from results based on *random* splits and used this to argue in favor of using random splits. While perhaps less common, random splits are already used in probing (Elazar and Goldberg, 2018), interpretability (Pörner et al., 2018), as well as core NLP tasks (Yu et al., 2019; Geva et al., 2019).[1]

Gorman and Bedrick (2019) focus on whether there is a significant performance difference δ
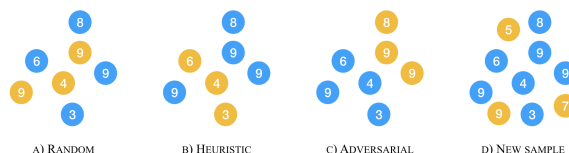

Figure 1: Data splitting strategies. Each ball corresponds to a sentence represented in (two-dimensional) feature space. Blue (dark)/orange (bright) balls represent examples for training/test. Numbers represent sentence length. Heuristic splits can, e.g., be based on sentence length; adversarial splits maximize divergence.

between systems S_1 and S_2; $\mathcal{M}(G_{test}, S_1) - \mathcal{M}(G_{test}, S_2)$, in their notation. They argue McNemar's test (Gillick and Cox, 1989) or bootstrap (Efron, 1981) can establish that $\delta \neq 0$, using random splits to sample from G_{test}. This, of course, relies on the assumption that data is representative, i.e., was sampled i.i.d. (Wolpert, 1996).

In reality, what Gorman and Bedrick (2019) call *the true difference in system performance*, i.e., $\delta = \mathcal{M}(G_{test}, S_1) - \mathcal{M}(G_{test}, S_2)$, is the system difference on *data that users would expect the systems to work well on* (see §2 for practical examples) – and not just on the corpus that we have annotations for. Our corpus-based estimates of δ can in fact be very misleading, i.e., very different from performance on new samples of data. In this paper, we investigate *how* misleading our estimates can be: We show that random splits consistently over-estimate performance at test time. This favors systems that overfit. We investigate alternatives across a heterogeneous set of NLP tasks. Based on our experiments, our answer to community-wide overfitting to standard splits is *not* to use random splits but to collect more diverse data with different biases – or if that is not feasible, split your data in adversarial, not random, ways. In general, we observe that estimates of test time error are worst for random splits, slightly better for standard splits

---

[1] See also many of the tasks in the SemEval evaluation campaigns: http://alt.qcri.org/semeval2020/

Task	Benchmark	Source/Domain	New Samples
POS TAGGING	WSJ	News	Xinhua
PROBING	SentEval	Toronto BC	Gutenberg
EMOJIS	S140	Twitter	S140*
QE	WMT 2016	IT	WMT 2018
HEADLINES	Gigaword	News	Gigaword*
NEWS	UCI	News	UCI*

Table 1: Data used in our experiments. *: We time slice the original data to create different samples.

(if those exist), better for heuristic and adversarial splits, but error still tends to be higher on new (in-domain) samples; see Figure 1.

Our results not only refute the hypothesis that δ can be estimated using random splits (Gorman and Bedrick, 2019),[2] but also the covariate shift hypothesis (Shimodaira, 2000; Shah et al., 2020) that δ can be estimated using reweightings of the data. While biased splits are useful in the absence of multiple held-out samples, and have been proposed before (Karimi et al., 2015),[3] they often overestimate performance in the wild. Our code is made publicly available at `https://github.com/google-research/google-research/talk_about_random_splits`.

2 Experiments

We consider 7 different NLP tasks: POS tagging (like Gorman and Bedrick (2019)), two sentence representation probing tasks, headline generation, translation quality estimation, emoji prediction, and news classification. We experiment with these tasks, because they a) are diverse, b) have not been subject to decades of community-wide overfitting (with the exception of POS tagging), and c) three of them enabled temporal splits (see Appendix §A.5).

Data splits The datasets which we will use in our experiments are presented in Table 1. For all seven tasks, we will present results for standard splits when possible (POS, PROBING,QE, HEADLINES), random splits, heuristic and adversarial splits, as well as on new samples. In the case of EMOJIS, HEADLINES and NEWS, which are all time-stamped datasets, we leave out historically

more recent data as our new samples. All **new samples** are in-domain samples of data where models are supposed to generalize, i.e, samples from similar text sources.[4] This is a key point: These are *samples that any end user would expect decent NLP models to fair well on*. Examples include a sample of newspaper articles from newspaper A for a POS tagger trained on articles from newspaper B; tweets sampled the day after the training data was sampled; or news headlines sampled from the same sources, but a year later.

We resample **random** splits multiple times (3-10 per task) and report average results. The **heuristic** splits are obtained by finding a sentence length threshold and putting the long sentences in the test split. We choose a threshold so that approximately 10% of the data ends up in this split. The idea of checking whether models generalize to longer sentences is not new; on the contrary, this goes back, at least, to early formal studies of recurrent neural networks, e.g., Siegelmann and Sontag (1992). In the §A.3, we present a few experiments with alternative heuristic splits, but in our main experiments we limit ourselves to splits based on sentence length.

Finally, the **adversarial** splits are computed by approximately maximizing the Wasserstein distance between the splits. The Wasserstein distance is often used to measure divergence between distributions (Arjovsky et al., 2017; Tolstikhin et al., 2018; Shen et al., 2018; Shah et al., 2018), and while alternatives exist (Ben-David et al., 2006; Borgwardt et al., 2006), it is easy to compute and parameter-free. Since selecting *the* worst-case split is an NP-hard problem (e.g., by reduction of the knapsack problem), we have to rely on an approximation. We first compute a ball tree encoding the Wasserstein distances between the data points in our sample. We then *randomly* select a centroid for our test split and find its k nearest neighbors. Those k nearest neighbors constitute our test split; the rest is used to train and validate our model. We repeat these steps to estimate performance on worst-case splits of our sample. See §A.4 for an algorithm sketch. Random, heuristic, and adversarial results are averaged across five runs.

---

[2]Or cross-validation, as more recently proposed in Szymański and Gorman (2020). In this very interesting follow-up paper, about Bayesian inference of δ, the authors write that their "estimates are valid insofar as the data sets used to estimate the Bayesian models comprise a representative sample of a coherent population of data sets." Our results show how *off* this assumption is.

[3]Karimi et al. (2015) discuss temporal splits and splits based on neighbor-based heuristics that are similar in spirit to our worst-case splits.

---

[4]Domains are commonly defined as collections of similar text sources (Harbusch et al., 2003; Koehn and Knowles, 2017). In addition to using similar sources, we control for low $\mathcal{A}$-distance (Ben-David et al., 2006) by looking at separability; e.g., a simple linear classifier over frequent unigrams can distinguish between Penn Treebank development and test sections with an accuracy of 64%; and between the development and our new sample with an accuracy of 69%.

POS tagging We first consider the task in Gorman and Bedrick (2019), experiment with heuristic and adversarial splits of the original Penn Treebank (Marcus et al., 1993), and add the Xinhua section of OntoNotes 5.0[5] as our **New Sample**. Our tagger is NCRF^{++} with default parameters.[6]

Probing We also include two SentEval probing tasks (Conneau et al., 2018) with data from the Toronto Book Corpus: PROBING-WC (word classification) and PROBING-BSHIFT (whether a bigram was swapped) (Conneau et al., 2018). Unlike the other probing tasks, these two tasks do not rely on external syntactic parsers, which would otherwise introduce a new type of bias that we would have to take into account in our analysis. We use the official SentEval framework[7] and BERT (Devlin et al., 2019) as our sentence encoder. The probing model is a logistic regression classifier with L_2 regularization, tuned on the development set. As our **New Samples**, we use five random samples of the 2018 Gutenberg Corpus[8] for each task, preprocessed in the same way as Conneau et al. (2018).

Quality estimation We use the WMT 2014 shared task datasets for QUALITY ESTIMATION. Specifically, we use the Spanish-English data from Task 1.1: scoring for perceived post-editing effort. The dataset comes with a training and test set, and a second, unofficial test set, which we use as our **New Sample**. In the §A.2, we also present results training on Spanish-English and evaluating on German-English. We present a simple model that only considers the target sentence, but performs better than the best shared task systems: we train an MLP over a LASER sentence embedding (Schwenk et al., 2019) with the following hyper-parameters: two hidden layers with 100 parameters each and ReLU activation functions, trained using the Adam stochastic gradient-based optimizer (Kingma and Ba, 2015), a batch size of 200, and L_2 penalty of strength $\alpha = 0.01$.

Headline generation We use the standard dataset for headline generation, derived from the Gigaword corpus (Napoles et al., 2012), as published by Rush et al. (2015). The task is to generate a headline from the first sentence of a news article.

Our architecture is a sequence-to-sequence model with stacked bi-directional LSTMs with dropout, attention (Luong et al., 2015) and beam decoding; the number of hidden units is 128; we do not pre-train. Different from Rush et al. (2015), we use subword units (Sennrich et al., 2016) to overcome the OOV problem and speed up training. The ROUGE scores we obtain on the standard splits are higher than those reported by Rush et al. (2015) and comparable to those of Nallapati et al. (2016), e.g., ROUGE-1 of 0.321. As our **New sample**, we reserve 20,000 sentence-headline pairs each from the first and second halves of 2004 for validation and testing; years 1998-2003 are used for training. For all the experiments we report the error reduction in ROUGE-2 of the model over the IDENTITY baseline, which simply copies the input sentence (other ROUGE values are reported in the §A.1). In §5, we will explore how much of a performance drop on the fixed test set is caused by shifting the *training* data by only five years to the past.

Emoji prediction Go et al. (2009) introduce an emoji prediction dataset, collected from Twitter and is time-stamped. We use the 67,980 tweets from June 16 as our **New Sample**, and tweets from all previous days for the remaining experiments. For this task, we again train an MLP over a LASER embedding (Schwenk et al., 2019) with hyper-parameters: two hidden layers with 50 parameters each and ReLU activation functions, trained using the Adam stochastic gradient-based optimizer (Kingma and Ba, 2015), a batch size of 200, and L_2 penalty of strength $\alpha = 0.01$. See §5 for a discussion of temporal drift in this data.

News classification We use a UCI Machine Learning Repository text classification problem.[9] Our datapoints are headlines associated with five different news genres. We use the last year of this corpus as our **New Sample**. We sample 100,000 headlines from the rest and train an MLP over a LASER embedding (Schwenk et al., 2019) with the following hyper-parameters: two hidden layers with 100 parameters and ReLU activation functions, trained using the Adam stochastic gradient-based optimizer (Kingma and Ba, 2015), dynamic batch sizes, and L_2 penalty of strength $\alpha = 0.01$.

---

[5]https://catalog.ldc.upenn.edu/LDC2013T19
[6]github.com/jiesutd/NCRFpp
[7]github.com/facebookresearch/SentEval
[8]tinyurl.com/zyq3yvn

[9]tinyurl.com/yysysmtr

Task	Model	Splits				
		Standard	**Random**	**Heuristic**	**Adversarial**	**New Samples**
POS TAGGING	NCRF^{++}	0.961	0.962	0.960	0.944	**0.927**
PROBING-WC	BERT	0.520	0.527	**0.232**	0.250	0.279
PROBING-BSHIFT		0.800	0.808	0.695	0.706	**0.450**
HEADLINE GENERATION*	seq2seq	0.073	0.095	0.062	**0.040**	0.069
QUALITY ESTIMATION†		0.502	0.626	0.621	0.711	**0.767**
EMOJI PREDICTION	MLP-Laser	-	0.125	0.196	**-0.040**	0.091
NEWS CLASSIFICATION		-	0.681	0.720	0.634	**0.618**
MSE (**New Samples**)		0.179	0.030	0.015	0.011	-

Table 2: Error reductions over random baselines on **Standard** (original) splits, if available, **Random** splits (obtained using cross-validation), **Heuristic** splits resulting from a sentence length-based threshold, **Adversarial** splits based on (five) approximate maximizations of Wasserstein differences between splits, and on **New Samples**. We bold face the **lowest** error reduction, i.e., where results differ the most from the random baseline. **We see that standard and random splits consistently over-estimate *real* performance on new samples, which is sometimes even lower than performance on adversarial splits.** We also report the mean squared error (MSE) with respect to **New Samples**, which shows **Adversarial** estimates empirical error best. Note: *While annotator bias could explain POS tagging results, there is no annotator bias in the other tasks.* *: For HEADLINES we use an identity baseline. Scores are ROUGE-2; see §A.1 for more. †: For QUALITY ESTIMATION, we report RMSE. The WMT QE 2014 best system obtained RMSE of 0.64; our system is significantly better with 0.50 on the standard split.

3 Results

Our results are presented in Table 2. Since the results are computed on different subsamples of data, we report *error reductions over multinomial random* (or, for HEADLINES, *identity*) *baselines*, following previous work comparing system rankings across different samples (Søgaard, 2013). More formally, we present error reduction as $r = \frac{p_s - p_b}{1 - p_b}$, where p_s and p_b are the performances of the system at hand and the multinomial random baseline.

Our main observations are the following: (a) **Random splits (and standard splits) consistently under-estimate error on new samples.** The absolute differences between error reductions over random baselines for random splits and on new samples are often higher than 20%, and in the case of PROBING-BSHIFT, for example, the BERT model reduces 80% of the error of a random baseline when data is randomly split, but only 45% averaging over five samples of new data from the same domain. (b) Heuristic splits sometimes under-estimate error on new samples. Our heuristic splits in the above experiments are quite aggressive. We only evaluate our models on sentences that are longer than any of the sentences observed during training. Nevertheless for 5/7 tasks, this leads to more optimistic performance estimates than evaluating on *new* samples! (c) The same story holds for

adversarial splits based on approximate maximization of Wasserstein distances between training and test data. While adversarial splits are very challenging, results on adversarial splits are more optimistic than on new samples in 4/7 cases. Note the fact that random splits over-estimate real-life performance also leads to misleading **system rankings**. If, for example, we remove the CRF inference layer from our POS tagger, performance on our **Random** splits drops to 0.952; on the **New Sample**, however, performance is 0.930, which is significantly better than *with* a CRF layer.

Discussion In the spirit of earlier work (Sakaguchi et al., 2017; Madnani and Cahill, 2018; Gorman and Bedrick, 2019), we provide recommendations for future evaluation protocols: (i) In the absence of multiple held-out samples, using **biased splits** better approximates real-world performance and can help determine what data characteristics affect performance. (ii) Evaluating on **new samples** is superior and also enables significance testing across datasets (Demsar, 2006), providing confidence estimates. Several benchmarks already provide multiple, diverse test sets (e.g. Hovy et al., 2006; Petrov and McDonald, 2012; Williams et al., 2018); we hope more will follow. What explains the high variance across samples in NLP? One reason is the dimensionality of language (Bengio et al.,

2003), but in §A.5 we also show significant impact of temporal drift.

Conclusions We have shown that out-of-sample error can be hard to estimate from random splits, which tend to underestimate error by some margin, but even biased and adversarial splits sometimes underestimate error on new samples. We show this phenomenon across seven very different NLP tasks and provide practical recommendations on how to best bridge the gap between experimental practices and what is needed to produce truly robust NLP models that perform well in the wild.

Acknowledgments

References

Martín Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223. PMLR.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. 2006. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 4-7, 2006*, pages 137–144. MIT Press.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155.

Karsten M. Borgwardt, Arthur Gretton, Malte J. Rasch, Hans-Peter Kriegel, Bernhard Schölkopf, and Alexander J. Smola. 2006. Integrating structured biological data by kernel maximum mean discrepancy. In *Proceedings 14th International Conference on Intelligent Systems for Molecular Biology 2006, Fortaleza, Brazil, August 6-10, 2006*, pages 49–57.

Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single \$&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018,*

Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers, pages 2126–2136.

Janez Demsar. 2006. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, 7:1–30.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Bradley Efron. 1981. Nonparametric estimates of standard error: the jackknife, the bootstrap and other methods. *Biometrika*, 68.

Yanai Elazar and Yoav Goldberg. 2018. Adversarial removal of demographic attributes from text data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 11–21. Association for Computational Linguistics.

Mor Geva, Eric Malmi, Idan Szpektor, and Jonathan Berant. 2019. Discofuse: A large-scale dataset for discourse-based sentence fusion. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 3443–3455. Association for Computational Linguistics.

L. Gillick and Stephen J. Cox. 1989. Some statistical issues in the comparison of speech recognition algorithms. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '89, Glasgow, Scotland, May 23-26, 1989*, pages 532–535. IEEE.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *Processing*, 150.

Kyle Gorman and Steven Bedrick. 2019. We need to talk about standard splits. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2786–2791. Association for Computational Linguistics.

Karin Harbusch, Saša Hasan, Hajo Hoffmann, Michael Kühn, and Bernhard Schüler. 2003. Domain-specific disambiguation for typing with ambiguous keyboards. In *Proceedings of the 2003 EACL Workshop on Language Modeling for Text Entry Methods*, TextEntry '03, page 67–74, USA. Association for Computational Linguistics.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. OntoNotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, New York City, USA. Association for Computational Linguistics.

Sarvnaz Karimi, Jie Yin, and Jiri Baum. 2015. Evaluation methods for statistically dependent text. *Comput. Linguistics*, 41(3):539–548.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation, NMT@ACL 2017, Vancouver, Canada, August 4, 2017*, pages 28–39. Association for Computational Linguistics.

Jan Lukes and Anders Søgaard. 2018. Sentiment analysis under temporal shift. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 65–71, Brussels, Belgium. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Nitin Madnani and Aoife Cahill. 2018. Automated scoring: Beyond natural language processing. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 1099–1109. Association for Computational Linguistics.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank. *Comput. Linguistics*, 19(2):313–330.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gu̇Ï‡lçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Courtney Napoles, Matthew Gormley, and Benjamin Van Durme. 2012. Annotated Gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pages 95–100,

Montréal, Canada. Association for Computational Linguistics.

Slav Petrov and Ryan McDonald. 2012. Overview of the 2012 shared task on parsing the web.

Nina Pörner, Hinrich Schütze, and Benjamin Roth. 2018. Evaluating neural network explanation methods using hybrid documents and morphosyntactic agreement. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 340–350. Association for Computational Linguistics.

Shruti Rijhwani and Daniel Preotiuc-Pietro. 2020. Temporally-informed analysis of named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7605–7617, Online. Association for Computational Linguistics.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Keisuke Sakaguchi, Courtney Napoles, and Joel R. Tetreault. 2017. GEC into the future: Where are we going and how do we get there? In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications, BEA@EMNLP 2017, Copenhagen, Denmark, September 8, 2017*, pages 180–187. Association for Computational Linguistics.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, and Armand Joulin. 2019. Ccmatrix: Mining billions of high-quality parallel sentences on the WEB. *CoRR*, abs/1911.04944.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Darsh J. Shah, Tao Lei, Alessandro Moschitti, Salvatore Romeo, and Preslav Nakov. 2018. Adversarial domain adaptation for duplicate question detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1056–1063. Association for Computational Linguistics.

Deven Shah, H. Andrew Schwartz, and Dirk Hovy. 2020. Predictive biases in natural language processing models: A conceptual framework and overview. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020,*

Online, July 5-10, 2020, pages 5248–5264. Association for Computational Linguistics.

Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. 2018. Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 4058–4065. AAAI Press.

Hidetoshi Shimodaira. 2000. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244.

Hava T. Siegelmann and Eduardo D. Sontag. 1992. On the computational power of neural nets. In *Proceedings of the Fifth Annual ACM Conference on Computational Learning Theory, COLT 1992, Pittsburgh, PA, USA, July 27-29, 1992*, pages 440–449. ACM.

Anders Søgaard. 2013. Estimating effect size across datasets. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 607–611. The Association for Computational Linguistics.

Piotr Szymański and Kyle Gorman. 2020. Is the best better? Bayesian statistical model comparison for natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2203–2212, Online. Association for Computational Linguistics.

Ilya O. Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schölkopf. 2018. Wasserstein autoencoders. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

David H. Wolpert. 1996. The lack of A priori distinctions between learning algorithms. *Neural Comput.*, 8(7):1341–1390.

Xintong Yu, Hongming Zhang, Yangqiu Song, Yan Song, and Changshui Zhang. 2019. What you see is what you get: Visual pronoun coreference resolution in dialogues. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 5122–5131. Association for Computational Linguistics.

	ROUGE-1	ROUGE-2	ROUGE-L
Standard	0.080	0.073	0.097
Random	0.109	0.095	0.127
Heuristic	0.091	0.062	0.109
Adversarial	0.060	0.040	0.080
New Sample	0.067	0.069	0.091

Table 3: Error reduction as compared with an identity baseline (output as input) for three ROUGE metrics. Random is a five-fold cross-validation result.
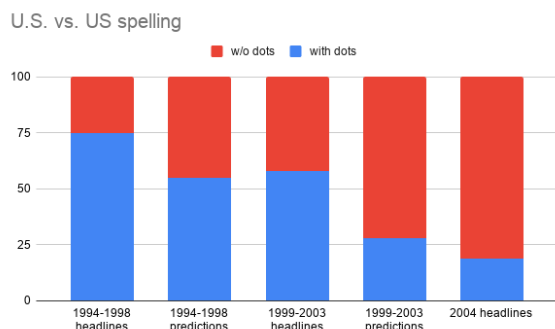

Figure 2: Proportions of *US* vs. *U.S.* spellings in the headlines for two training sets and the test set (2004) as well as in the two models' predictions on the test set.

A Appendices

We present supplementary details about two of our tasks in §A.1 and §A.2 and discuss variations over heuristic splits in §A.3. In §A.4, we present the pseudo-algorithm for how we compute adversarial splits, and finally, in §A.5, we present our results documenting temporal drift.

A.1 Headlines

Table 3 reports the error reduction in ROUGE-1, ROUGE-2 and ROUGE-L over the identity baseline (see §2) for the different data splits. The results are consistent with Table 2. Figure 2 gives more details on an interesting drift phenomenon, which contributed to the superior performance of the model trained on the most recent five years (1999-2003). Apparently, the dotless spelling of *U.S./US* ('United States') became more common over time. Consequently, the model trained on the 1999-2003 part generated *US* more frequently than the model trained on 1994-1998.

A.2 Quality Estimation

In the results above, we train and test our quality estimation regressor on Spanish-English from WMT QE 2014. We also ran a similar experiment where we used the German-English test data as our

New Sample. Here, we see a similar pattern to the one above: The RMSE on the **Standard** split was 0.630, which is slightly higher than for Spanish-English; with our **Heuristic** split, RMSE is 0.652; for **Adversarial**, it is 0.626 (which is slightly better than with standard splits), and on our **New Sample**, RMSE is 0.813.

A.3 Alternative Heuristic Splits

For both SentEval tasks we experimented with the following alternatives for heuristic splits.

Bootstrap Resampling Instead of cross-validation, a random split can be generated by bootstrap resampling. For this we randomly select 10% of the data as test set and then randomly sample (with replacement) a new training and dev set from the remaining examples.

Random Length As alternative to the length threshold heuristic in earlier experiments we randomly sample a length and select all examples having this length to be part of the test set. We repeat this procedure until approximately 10% of the data ends up in the test set. With this procedure we create 5 different test sets. We included this heuristic in order to see how fragile the probing setup is.

Rare Words Another alternative for heuristic splits is to use word frequency information. Here we assign those sentences containing at least one of the rarest words of the dataset to the test set. This way we end up again with approximately 10% of the data in the test set. Note that this way we create only 1 dataset, because it's not a random process.

Results Table 4 lists the results. While bootstrap resampling leads to slightly lower error reduction than cross-validation we decided to report the latter in the main part of this paper, because it is a more wide-spread way to randomly split datasets. Random Length results are comparable to standard splits results. The split based on word frequency (Rare Words) leads to considerable drop in both tasks. However, it is not as strong as the drop of the heuristic split (length threshold) in the main part of the paper.

A.4 Computing adversarial splits

We present the pseudo-algorithm of our implementation of approximate Wasserstein splitting in Algorithm 1. We also make the corresponding code available as part of our code repository for this paper.

Task	Model	Splits			
		Standard	**Bootstrap**	**Random Length**	**Rare Words**
PROBING-WC	BERT	0.520	0.504	0.554	0.339
PROBING-BSHIFT		0.800	0.807	0.798	0.731

Table 4: Error reductions over random baselines on **Standard** (original) splits, if available, **Bootstrap** splits, **Random Length** splits resulting from a sentence length-based separation, **Rare Words** splits based on word frequency.
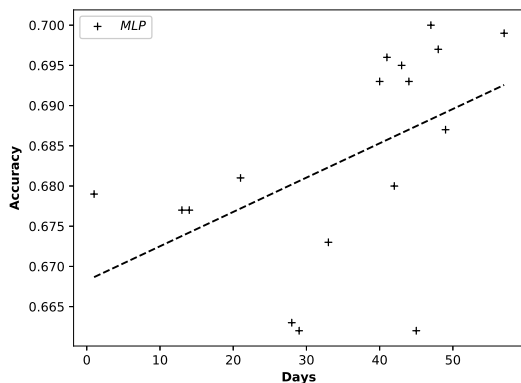

Figure 3: Temporal drift in emoji prediction. The correlation between temporal gap and performance is significant ($p < 0.05$).

Data: Dataset G_{train}
Result: Adversarial split: G_{train} and G_{test}
$T \leftarrow WassersteinBallTree(G_{train})$;
$G_{test} \leftarrow \{\langle x, y \rangle \sim G_{train}\}$;
$G_{train} \leftarrow G_{train}/\{\langle x, y \rangle\}$;
while $i \leq k$ **do**
$\quad \langle x_i, y_i \rangle = \min_{\langle x', y' \rangle} T(x, x')$;
$\quad G_{train} \leftarrow G_{train}/\{\langle x_i, y_i \rangle\}$;
$\quad G_{test} \leftarrow G_{test} \cup \{\langle x_i, y_i \rangle\}$;
$\quad i += 1$;
end
Algorithm 1: Computing adversarial splits

	ROUGE-1	ROUGE-2	ROUGE-L
Identity baseline	0.302	0.110	0.260
50% of 1994-2003	0.409	0.205	0.386
1994-1998	0.346	0.161	0.329
1999-2003	0.413	0.208	0.388

Table 5: Performance of three seq2seq models trained on different samples, evaluated on 2004 data.

A.5 The significance of drift

Some of our splits in the main experiments were based on slicing data into different time periods (HEADLINES, EMOJIS). Since temporal drift is a potential explanation for sampling bias, we analyze this in more detail here. We show that temporal drift is pervasive and leads to surprising drops in performance. We note, however, that temporal drift is not the only cause of sampling bias, of course. Since we have time stamps for two of our datasets we study these in greater detail. For similar studies of temporal drift, see Lukes and Søgaard (2018); Rijhwani and Preotiuc-Pietro (2020).

Headline generation Our headline generation data covers the years 1994 to 2004. Having reserved 20,000 sentence-headline pairs from the first half of 2004 for validation and the same amount from the second half for testing, we use 50% of the years 1994-2003 for training three models. The models' architectures and parameters are identical (same as in Sec. 3). The only difference is in what the models are trained on: (a) a random half, (b) the first, or (c) the second half of 1994-2003. The training data sizes are comparable (1.63-1.76M), the publisher distributions (AFP, APW, CNA, NYT or XIN) are also similar. Hence, the models are expected to perform similarly on the same test set.

As Table 5 indicates, shifting the training data by five years to the past results in a big performance drop. Sampling training data randomly or taking the most recent period produces models with

similar ROUGE scores, both much better than the identity baseline. However, about half of the gap to the identity baseline disappears when older training data is taken. In the §A.1, we give an example of temporal drift in the HEADLINES data: *US* largely replaces *U.S.* in the newer training set and the test set.

Emoji prediction   For emoji prediction, Go et al. (2009) provide data for a temporal span of 62 days. We split the data into single days and keep the splits with more than 25,000 datapoints in which both classes are represented. We use the last of these, June 16, as our test sample and vary the training data from the first day to the day before June 16. Figure 3 (left) visualizes the results.