Mark-Evaluate: Assessing Language Generation using Population Estimation Methods

Gonçalo Mordido
Hasso Plattner Institute
Potsdam, Germany
`goncalo.mordido@hpi.de`

Christoph Meinel
Hasso Plattner Institute
Potsdam, Germany
`christoph.meinel@hpi.de`

Abstract

We propose a family of metrics to assess language generation derived from population estimation methods widely used in ecology. More specifically, we use mark-recapture and maximum-likelihood methods that have been applied over the past several decades to estimate the size of closed populations in the wild. We propose three novel metrics: $ME_{Petersen}$ and $ME_{CAPTURE}$, which retrieve a single-valued assessment, and $ME_{Schnabel}$ which returns a double-valued metric to assess the evaluation set in terms of quality and diversity, separately. In synthetic experiments, our family of methods is sensitive to drops in quality and diversity. Moreover, our methods show a higher correlation to human evaluation than existing metrics on several challenging tasks, namely unconditional language generation, machine translation, and text summarization.

1 Introduction

Population estimation methods have been widely used in ecology to study the development of species over the last several decades (Krebs and others, 1989). Existing population estimation methods focus on open populations, where births, deaths, and migrations are taken into account, or closed population methods, where the population is assumed to remain static over the population estimation study. In this work, we focus on closed population methods and study how their respective population estimates can be used to evaluate an evaluation set of generated samples, given a reference set of real samples. In this work, samples are either contextualized word or sentence embeddings. We study two mark-recapture methods, namely the Petersen (Ricker, 1975) and the Schnabel (Schnabel, 1938) estimators, where samples are captured and marked, released, and recaptured. The number of such different samples is then used to estimate the population size. We further use CAPTURE (Otis et al., 1978), a maximum-likelihood method, which uses the number of marked samples over multiple captures for such estimation.

Accurate evaluation of generated data is essential to correctly measure in what degrees we can improve the overall generation process. Depending on the use case, single-valued metrics may suffice to assess specific conditional language generation tasks, such as machine translation and text summarization, where we are interested in evaluating the similarities of a generated translation or summary to a specific reference translation or summary, respectively. On the other hand, on unconditional language generation, for example, it may be useful to have separate measures for the diversity and quality of the generated set, enabling the identification of possible shortcomings of our generation system and try to fix it accordingly. This has been an active area of generative models research (Goodfellow et al., 2014), with several works focusing on stimulating diversity while maintaining the overall sample quality (Srivastava et al., 2017; Lin et al., 2018; Mordido et al., 2018; Mordido et al., 2020; Sauder et al., 2020).

Mark-Evaluate (ME) is a family of 3 novel language evaluation methods based on the above population estimation methods: $ME_{Petersen}$ and $ME_{CAPTURE}$ retrieve a single-valued metric to assess an evaluation set, while $ME_{Schnabel}$ returns a double-valued metric, separately measuring the quality and diversity of the evaluation set. Our main contributions can be listed as follows: (i) Proposal of 3 novel language metrics

(Section 3) that are sensitive to mode collapse (Section 4.1) and quality detriment (Section 4.2) and show a high correlation to human evaluation on challenging text generation tasks, such as unconditional language generation (Section 5), machine translation (Section 6.1) and text summarization (Section 6.2). (ii) In-depth study of the language assessment capability of popular existing metrics, *i.e.* FID (Heusel et al., 2017), PRD (Sajjadi et al., 2018) and IMPAR (Kynkäänniemi et al., 2019), primarily used to evaluate image generation in the past. (iii) Usage of contextual information and different levels of granularity to assess language, by using either contextualized word (Sections 6.1 and 6.2) or sentence embeddings (Sections 4, 5) derived from BERT (Devlin et al., 2019). (iv) Code for the reproducibility of the results will be publicly available.

2 Related work

While acknowledging the importance of traditional evaluation metrics, such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004) and METEOR (Banerjee and Lavie, 2005), we will focus on the new trend of unsupervised methods that use embedding representations of pre-trained models to assess a set of evaluation samples. Our family of methods analyzes the data manifold to assess the evaluation set by using *k-nearest neighbors* to determine the capture volume. Several methods have been recently proposed to assess data generation using topological information, however, they were primarily intended to assess image generation (Sajjadi et al., 2018; Khrulkov and Oseledets, 2018; Kynkäänniemi et al., 2019; Niedermeier et al., 2020). In this work, we investigate the performance of such methods in the text domain, analyzing their behavior on synthetic experiments and their correlation with human evaluation.

Precision and recall for distributions (PRD) was proposed by Sajjadi et al. (2018) and uses of *k-means* (MacQueen and others, 1967) to build histograms of the discrete reference and evaluation distributions over the clusters' centers. The evaluation distribution is then assessed in terms of relative probability densities. Precision is obtained by calculating the probability of an evaluation sample falling within the reference distribution's support. On the other hand, recall is retrieved by calculating the probability of a reference sample falling within the evaluation distribution's support.

Kynkäänniemi et al. (2019) suggested several improvements to the above method, which we call improved precision and recall (IMPAR). First, instead of *k-means*, they proposed to use *k-nearest neighbors* to approximate the reference and evaluation manifolds by building a hypersphere around each sample to its k-th nearest neighbor. Second, they simplify PRD's notions of precision and recall, by calculating the probability of an evaluation sample to fall within at least one reference sample's hypersphere, and vice-versa, respectively. The proposed manifold approximations by the usage of hyperspheres present a simple, yet effective way of representing the reference and evaluation manifold in an explicit, non-parametric way. We build upon this idea and use identical hyperspheres to determine the capture volume used by our different estimators to estimate the population size.

Fréchet Inception Distance or FID (Heusel et al., 2017) is a widely used single-valued metric that assesses data similarity by calculating the distance between the reference and evaluation distributions. Even though originally proposed for the image domain, Semeniuta et al. (2018) adapted FID to evaluate text generation by getting vector representations from InferSent (Conneau et al., 2017), instead of Inception-V3 (Szegedy et al., 2016). Even though this metric precedes PRD and IMPAR, FID is still commonly used to assess generative models in the image and text domain.

As previously mentioned, we study both the usage of sentence embeddings, derived from SBERT (Reimers and Gurevych, 2019), as well as contextualized word embeddings from BERT (Reimers and Gurevych, 2019), which have been recently shown to improve language assessment, both in a supervised (Mathur et al., 2019; Sellam et al., 2020) and unsupervised manner (Zhao et al., 2019; Zhang* et al., 2020). More specifically, BERTScore measures precision and recall from a reference and evaluation text by calculating the required transport of each word of a given text to the most semantically similar word in the other text. On the other hand, MoverScore measures the semantic distance between two texts by calculating the minimum transport required between the reference and evaluation texts. These metrics are ideal for conditional text generation, such as machine translation and text summarization, where an evaluation text should match a given reference text.

Figure 1: Each sample $s \in S$ and $s' \in S'$ is represented as a blue and red circle, respectively. Marked samples are represented by filled circles. In this illustration, hyperspheres reach to each sample's nearest neighbor of the same set, *i.e.* $K = 1$. For $\text{ME}_{\text{Petersen}}$ and $\text{ME}_{\text{Schnabel}}$, we first capture and mark all samples inside any hypersphere of s. Then, for $\text{ME}_{\text{Petersen}}$, we count the number of marked samples or recaptures inside any hypersphere of s'. For $\text{ME}_{\text{Schnabel}}$, we perform a similar process iteratively, marking and recapturing samples inside the hypersphere of each s', resulting in all samples being marked in the end. On the other hand, $\text{ME}_{\text{CAPTURE}}$ captures and marks samples inside each hypersphere of s and s'.
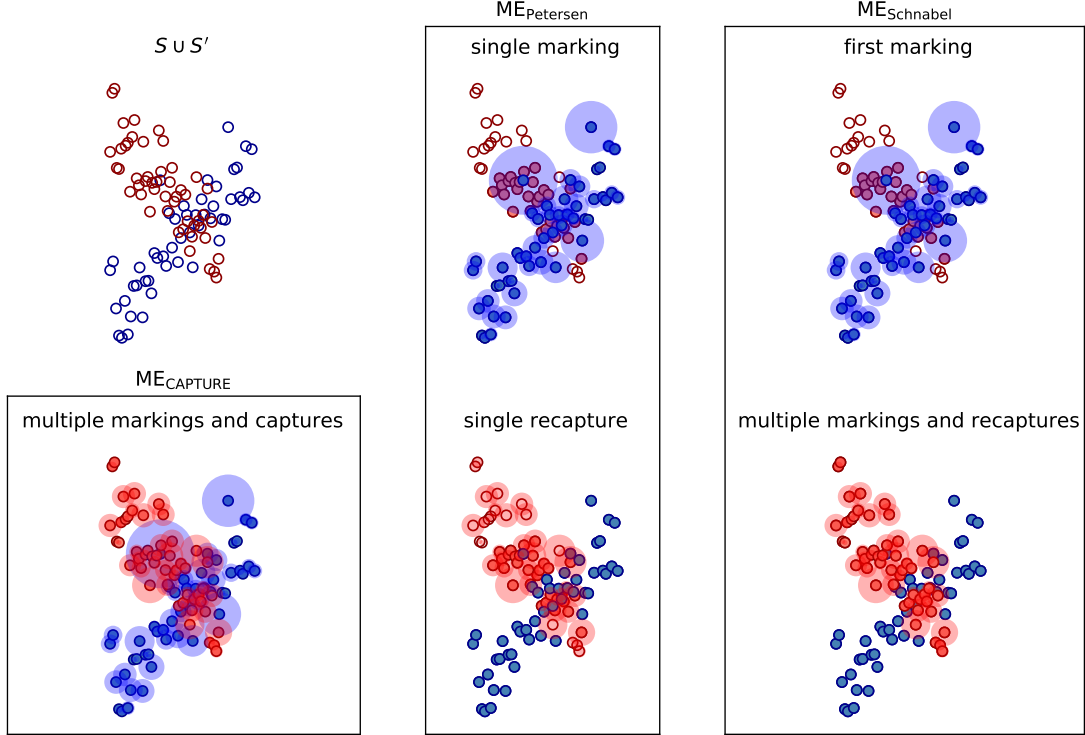
3 Mark-Evaluate

In this work, we consider population estimation methods for closed populations, where the true population size remains constant throughout the estimation study. In our use case, our population consists of two sets, namely a reference set S_r and an evaluation set S_e. The true population size (P) is then known *a priori* and represents the total number of samples in the two sets: $P = |S_r| + |S_e|$. Given an estimated population size ($\widehat{P}$) from one of the used estimators, we measure the accuracy loss (A) as follows:

$$A(P, \widehat{P}) = \max\left(\left|\frac{\widehat{P} - P}{P}\right|, 1\right), \tag{1}$$

with a low accuracy loss, *i.e.* $A(P, \widehat{P}) \approx 0$, representing a good population estimate, and a high accuracy loss, *i.e.* $A(P, \widehat{P}) \approx 1$, otherwise. Our population estimation methods assume all samples to have an equal chance of capture, which is influenced by our capture volumes: hyperspheres that reaches each reference or evaluation sample's k-th nearest reference or evaluation neighbor, respectively. Hence, if evaluation samples tend not to be inside any reference sample's hypersphere and vice-versa, the population estimate will likely be poor due to the lack of captured samples in the estimation study.

Our methods can be separated into three categories: single marking and recapture ($\text{ME}_{\text{Petersen}}$), multiple markings and recaptures ($\text{ME}_{\text{Schnabel}}$) and multiple markings and captures ($\text{ME}_{\text{CAPTURE}}$). Let us consider two sets of samples S and S', where each sample is a contextualized word embedding or a sentence embedding derived from BERT, depending on the task. Figure 1 illustrates our family of methods.

Adapting Kynkäänniemi et al. (2019)'s formulations, we define a binary function f that returns whether a sample $s' \in S'$ lays inside any capture volume or hypersphere of a sample $s \in S$:

$$f(s', S) = \begin{cases} 1, & \text{if} \quad ||s' - s||_2 \leq ||s - \text{NN}_k(s, S)[-1]||_2 \quad \text{for at least one} \quad s \in S \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

where $\text{NN}_k(s, S)$ returns an ordered set containing s and its k-nearest neighbors in the set S, in ascending order of Euclidean distances to s. Hence, $\text{NN}_k(s, S)[-1]$ represents the k'th nearest neighbor of s. We may refer to individual samples in S and S' as $\{s_1, \ldots, s_{|S|}\}$ and $\{s'_1, \ldots, s'_{|S'|}\}$, respectively.

The Petersen estimator (Ricker, 1975), relies on a single marking step and a single recapture step. It merely assumes that the ratio of marked samples (M) in the marking step and the population size (P) is equivalent to the ratio of recaptured samples (R) and captured samples (C) in the recapture step. The population size estimate $(\widehat{P}_{\text{Petersen}})$ is then calculated as follows:

$$\widehat{P}_{\text{Petersen}}(S, S') = \frac{C(S, S')M(S, S')}{R(S, S')}. \tag{3}$$

During the marking step, we mark all samples inside at least one hypersphere of s: $M(S, S') = |S| + \sum_{s' \in S'} f(s', S)$. During the recapture step, we do the opposite, marking all samples inside at least one hypersphere of s': $C(S, S') = |S'| + \sum_{s \in S} f(s, S')$. Additionally, in the recapture step, we count the number of captured samples that are already marked from the marking step, *i.e.* the recaptured samples. This corresponds to the number of samples in S' inside at least one hypersphere of s as well as the number of samples in S inside at least one hypersphere of s'': $R(S, S') = \sum_{s' \in S'} f(s', S) + \sum_{s \in S} f(s, S')$.

The Petersen estimator was extended by Schnabel (1938) to incorporate multiple markings and recaptures. The population size estimate $(\widehat{P}_{\text{Schnabel}})$ is calculated from T consecutive Petersen estimates:

$$\widehat{P}_{\text{Schnabel}}(S, S') = \frac{C_T(S, S')M_T(S, S')}{R_T(S, S')}, \tag{4}$$

The set of marked samples at each iteration $t \in \{1, \ldots, T\}$, can be defined recursively as:

$$M(t, S, S') = \begin{cases} S \cup \{s' \in S' | f(s', S) = 1\}, & \text{if } t = 1, \\ S \cup S', & \text{if } t = T, \\ M(1, S, S') \cup \left(\bigcup_{i=1}^{t-1} \text{NN}_k(s'_i, S') \right), & \text{otherwise.} \end{cases} \tag{5}$$

$\text{ME}_{\text{Schnabel}}$'s first marking step is identical to $\text{ME}_{\text{Petersen}}$'s single marking step $(M(1, S, S') = M(S, S'))$, with all samples in S as well as samples in S' that are inside at least one hypersphere of s being marked. By the final marking step, all samples will be marked since we iterated through all of them: $M_T(S, S') = |M(T, S, S')|$. For the other iterations, $1 < t < T$, samples in S' that are captured, *i.e.* are k-nearest neighbors of the s' being iterated, but are not yet marked, are added to the marked set.

After all recapture steps, which excludes the first marking step, the number of captured samples will be the number of samples in S' and their respective k'th nearest neighbors as well as samples in S that are inside the hypersphere of each s': $C_T = (K + 1) * |S'| + \sum_{s' \in S'} \sum_{s \in S} f(s, \text{NN}_k(s', S'))$. Since all samples in S have been marked in the first marking step, the number of total recaptures is the number of samples in S inside the hypersphere of each s' as well as the number of k-nearest neighbors of the iterated s' that have already been marked: $R_T(S, S') = \sum_{i=1}^{|S'|} \sum_{j=1}^{|S|} \left(f(s_j, \text{NN}_k(s'_i, S')) + |M(i, S, S') \cap \text{NN}_k(s'_i, S')| \right)$.

Both $\text{ME}_{\text{Petersen}}$ and $\text{ME}_{\text{Schnabel}}$ are mark-recapture methods since they rely on marking and recapturing information to estimate the population size. We further used a maximum log-likelihood method: the model null from Program CAPTURE (Otis et al., 1978). By considering the total number of marked

Figure 2: Effects of using different capture volumes, *i.e.* changing the number of neighbors (K), in the population size estimated of each estimator (left) and our respective method's score (right).
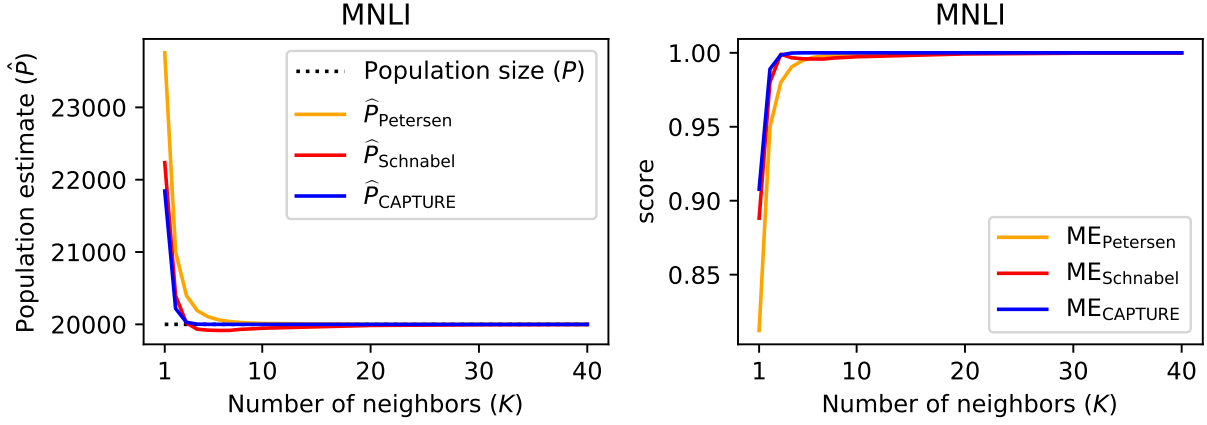
samples (M_T) and the total number of captures (C_{total}) over T iterations, with $T = |S \cup S'|$, we iterate through several provisional population estimates ($P_{\text{CAPTURE}} \in \mathbb{N}_{\geq M}$) and compute their log-likelihood:

$$L_n(P_{\text{CAPTURE}}; S, S') = \ln\left(\frac{P_{\text{CAPTURE}}!}{(P_{\text{CAPTURE}} - M_T(S, S'))!}\right) + C_{total}(S, S') \times ln\left(C_{total}(S, S')\right)$$

$$+ \left(TP_{\text{CAPTURE}} - C_{total}(S, S')\right) \times \ln\left(TP_{\text{CAPTURE}} - C_{total}(S, S')\right) - (TP_{\text{CAPTURE}}) \ln(TP_{\text{CAPTURE}}). \tag{6}$$

The total number of captures corresponds to the number of samples in S and S' and their respective neighbors, as well as the number of samples in S inside the hypersphere of a given s' and vice-versa:
$$C_{total}(S, S') = \sum_{s \in S} \sum_{s' \in S'} (f(s', \text{NN}_k(s, S)) + |\text{NN}_k(s, S)|) + \sum_{s' \in S'} \sum_{s \in S} (f(s, \text{NN}_k(s', S')) + |\text{NN}_k(s', S')|).$$
The final population estimate ($\widehat{P}_{\text{CAPTURE}}$) is then the estimate that maximizes Equation 6:

$$\widehat{P}_{\text{CAPTURE}}(S, S') = \underset{P_{\text{CAPTURE}}}{\text{argmax}} \, \widehat{L_n}(P_{\text{CAPTURE}}; S, S'). \tag{7}$$

Our family of methods uses the accuracy loss of each estimator to compute their scores as follows:

$$\text{ME}_{\{\text{Petersen, Schnabel, CAPTURE}\}}(S, S') = 1 - A\left(P, \widehat{P}_{\{\text{Petersen, Schnabel, CAPTURE}\}}(S, S')\right). \tag{8}$$

Note that, due to its iterative nature, $\text{ME}_{\text{Schnable}}$ may be used to separately assess the quality and diversity of an evaluation set S_e given a reference set S_r. More specifically, quality may be calculated by $\text{ME}_{\text{Schnable}}(S_r, S_e)$, whereas diversity may be measured by $\text{ME}_{\text{Schnable}}(S_e, S_r)$. On the other hand, $\text{ME}_{\text{Petersen}}$ and $\text{ME}_{\text{CAPTURE}}$ are single-valued metrics, since $\text{ME}_{\text{Petersen}}(S_r, S_e) = \text{ME}_{\text{Petersen}}(S_e, S_r)$ and $\text{ME}_{\text{CAPTURE}}(S_r, S_e) = \text{ME}_{\text{CAPTURE}}(S_e, S_r)$. We refer to the Appendix for theoretical discussions.

To study the effects of different capture volumes, determined by different K, we used SBERT to get the sentence embeddings of 10k training sentences from MNLI (Williams et al., 2017) as the reference set, and 10k validation sentences as the evaluation set. Results are shown in Figure 2, with $K \in \{1, \ldots, 40\}$. We observe that as K increases, the population size estimated by all estimators converges to the true population size. In turn, the scores of our family of methods also converge to their maximum value of 1.

4 Synthetic experiments

To simulate drops in quality and diversity, we used the MNLI dataset (Williams et al., 2017), which consists of 433k sentence pairs annotated with one out of 5 possible topics. Since we were interested in

Figure 3: Mode collapse experiment where sentences from certain topics are dropped from the evaluation set. Quality (full lines) is expected to remain constant while diversity (dotted lines) is expected to drop as mode collapse aggravates.
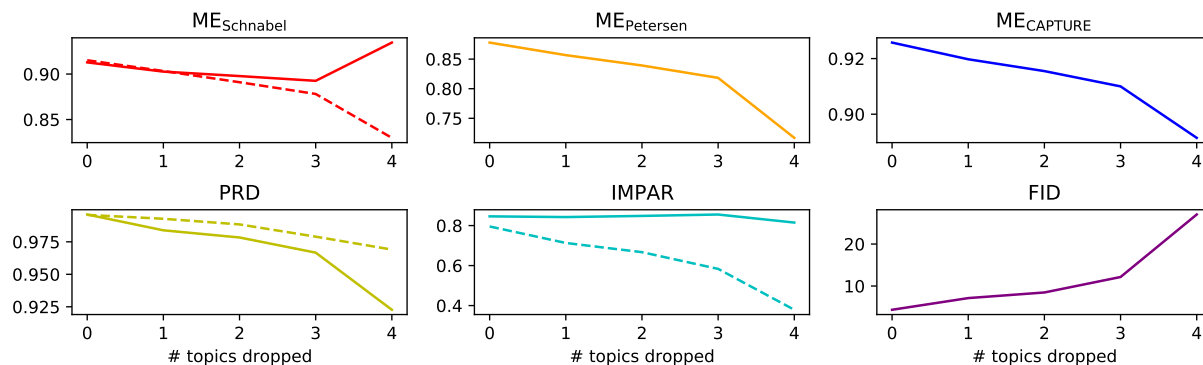
the sentence-pair information for these experiments, we treated each sentence independently. Our reference set consists of sentences from the training set, whereas our evaluation set has sentences from the validation set. We kept the size of the reference and evaluation sets equal throughout our experiments to reduce possible method instabilities regarding sample size. We follow the experiments in Semeniuta et al. (2018) and simulate diversity loss by dropping sentences from certain topics (Section 4.1), whereas quality detriment is induced by swapping the words of each sentence (Section 4.2). For both experiments, we use SBERT embeddings from BERT-base pre-trained on SNLI (Bowman et al., 2015) and MNLI (Williams et al., 2017) datasets (*'bert-base-nli-mean-tokens'* [1]).

4.1 Mode collapse

To evaluate mode collapse, we dropped the sentences from specific topics from the evaluation set, containing sentences from all the available topics. Thus, the evaluation set only contains sentences from a subset of topics. What differs at each step is the number of topics included in the evaluation set: for example, dropping one topic means that the evaluation set only contains samples from the rest of the four available topics. The reference set remained unaltered throughout this process. We used 4k reference and 4k evaluation samples throughout this experiment.

We expect quality assessments to remain constant and diversity assessments to drop as fewer topics are represented in the evaluation set. For single-metric methods, we expect a detriment of the overall score throughout the mode dropping process. Results are presented in Figure 3, where we observe that our family of methods displays the expected behavior. IMPAR and FID also show expected performance (note that higher FID is worse since it represents a distance from the reference and evaluation distributions). On the other hand, PRD's quality assessment or precision drops significantly as mode collapse aggravates, which is not expected since the quality of the evaluation set is not affected in this experiment. We further observe that all methods show high sensitivity when only one topic is represented in the evaluation set. For example, when 4 topics are dropped, IMPAR's quality assessment shifts by ≈ 0.4, while $ME_{Schnabel}$ and $ME_{Schnabel}$ shifts by ≈ 0.5. Hence, both methods show similar variance, despite the visualization contrast originated from different y-scales.

4.2 Word swap

To evaluate quality detriment, we swapped the words of each sentence in the evaluation set with a certain swap probability. Similarly to the mode collapse experiment, the reference set remains constant throughout this study. We used 10k reference and 10k evaluation samples. Semeniuta et al. (2018) showed that sentence embeddings derived from InferSent (Conneau et al., 2017) and Transformers (Vaswani et al., 2017) models were unable to detect similar quality perturbations. However, we observe that BERT

---
[1]https://github.com/UKPLab/sentence-transformers

1968

Figure 4: Swap experiment where words from each sentence in the evaluation set are swapped with a certain swap probability. Quality (full lines) is expected to drop while diversity (dotted lines) is expected to remain constant as the swap probability increases.
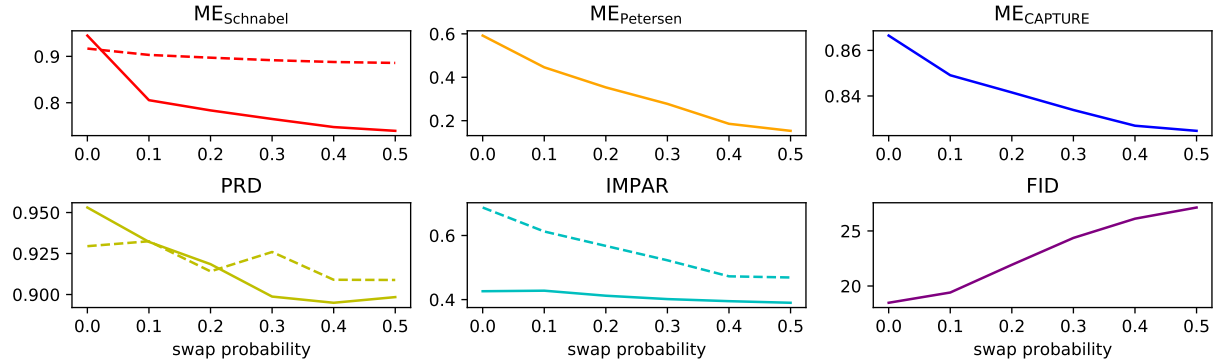
embeddings can capture such quality detriment, observed by the variance of the scores of all the tested methods.

For this experiment, precision is expected to drop, while recall should remain constant. The overall score of single-valued metrics should deteriorate as the swap probability increases. Results are presented in Figure 4. Both the quality and diversity assessments of $ME_{Schnabel}$ show the expected behavior, similarly to our single-metrics and FID. On the other hand, the recall or diversity assessment of both PRD and IMPAR drops unexpectedly. Moreover, IMPAR's quality or precision does not drop as significantly at higher swap probabilities, which is not desirable.

5 Language generation

We further assessed the text generated by ten different language generation models presented in Cífka et al. (2018). The models include a traditional language model and several types of autoencoders, namely variational, adversarial, adversarially regularized, and plain autoencoders. We used the human ratings assigned to each model's fluency presented in their work to study the correlation of our family of methods and other tested metrics to human evaluation. Reverse and forward cross-entropy, *i.e.* Reverse CE and Forward CE, have been commonly used to assess text generation in the past (Cífka et al., 2018; Semeniuta et al., 2018; Zhao et al., 2018). The reported Reverse CE and Forward CE results were taken from Cífka et al. (2018), obtained by training a language model on English Gigaword (Napoles et al., 2012). We refer to Cífka et al. (2018) for additional details.

We used SBERT embeddings from BERT-large trained on SNLI and MNLI datasets (*'bert-large-nli-mean-tokens'*) since they achieved the best-reported performance in Reimers and Gurevych (2019). Note that, since human evaluation is only related to each model's fluency, we only report the quality assessment scores for $ME_{Schnabel}$, PRD, and IMPAR. For our family of methods, as well as PRD and IMPAR, we iterate through K values until correlation drops and present the results with the best K of each method. Table 1 shows the Pearson p, Kendall k, and Spearman p correlations to human evaluation.

Correlations	FORWARD CE	REVERSE CE	FID	PRD	IMPAR	$ME_{Schnabel}$	$ME_{Petersen}$	$ME_{CAPTURE}$
Pearson r	0.606	0.440	0.902	0.830	0.745	**0.917**	0.872	**0.902**
Kendall k	0.556	0.333	0.867	0.822	0.778	**<u>0.911</u>**	**<u>0.911</u>**	0.867
Spearman p	0.697	0.491	0.964	0.939	0.903	**<u>0.976</u>**	**<u>0.976</u>**	0.964

Table 1: Absolute correlations to human evaluation regarding the fluency of 10 different models with the best K. Best scores for each correlation are underlined. Bold values represent the correlations where our methods outperform or match all of the other methods' performance.

Figure 5: When using contextualized word embeddings in our family of methods, each word is represented by five samples corresponding to the embeddings of the last five layers of BERT. We used the same procedure as Mark-Evaluate for PRD and IMPAR. On the other hand, MoverScore uses p-means to aggregate the information of the last five embeddings, and BERTScore uses the embeddings of a given layer for each word.
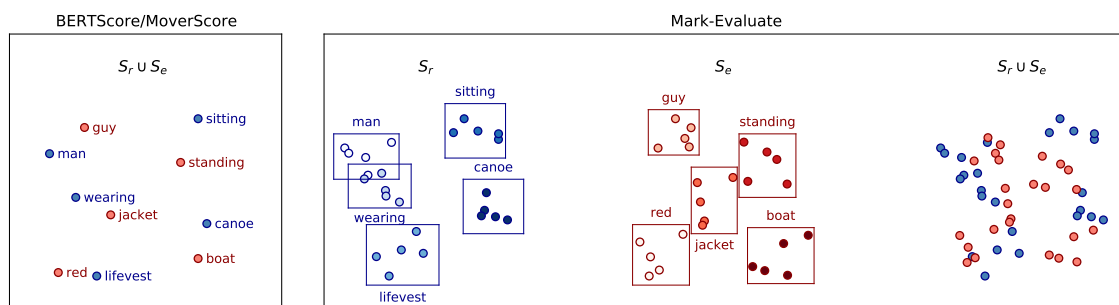
Overall, our family of methods achieves the highest correlations to human evaluation. Note that despite being outperformed by FID, ME$_{Petersen}$ still outperforms PRD and IMPAR across all correlations. Additional results with default K for our family of methods, PRD, and IMPAR, as well as a comparison with InferSent and different SBERT embeddings, are provided in the Appendix.

6 Contextualized word embeddings

We will now shift our focus to conditional language generation under finer-grained representations, *i.e.* contextualized word embeddings. We used embeddings from BERT-base fine-tuned on MNLI, identically to Zhao et al. (2019). For a fair comparison, we used the same embedding representations for all the methods in the following experiments. Due to the likely imbalance of reference and evaluation samples, we only report the quality assessment or precision of double-valued metrics. Similarly to Section 5, we report the results with the best K. Additional results with default K can be found in the Appendix.

Using the information of the last layers of BERT has been shown to help in several downstream tasks (Liu et al., 2019a). This has also been shown for language assessment, observed by, for example, the fact that the best performing layers of BERTScore are often latter layers (Zhang* et al., 2020). MoverScore extends this thinking and aggregates the representations of the last five layers of BERT with p-means. For our methods, instead of aggregating or routing this information, we use the vector representation from the last five layers for each specific word. Thus, each word has five representations, defined as five samples, in our scheme. See Figure 5 for an illustration of this process. This also allows us to produce a better population estimate in the end due to the increase of the sample size.

6.1 Machine translation

We start by assessing system-level machine translations from the WMT17 metrics task (Bojar et al., 2017). We evaluated the different methods on the five language pairs provided by Zhao et al. (2019)'s implementation [2]. Namely, we assess translations from Czech (cs), German (de), Russian (ru), Turkish (tr), and Chinese (zh) to English (en). Each language pair has around 3k reference with the respective evaluation translations from multiple systems (the number of systems for each language pair varies).

Pearson (r) correlations with human evaluation are presented in Table 2. Our family of metrics outperforms all the rest in several language pair translations. Moreover, our metrics show the highest correlation to human evaluation when considering the average correlation across all language pairs. BERTScore results were calculated using the embeddings from the last fifth layer of the aforementioned BERT model.

---

[2]https://github.com/AIPHES/emnlp19-moverscore

Translations	BERTScore	MoverScore	PRD	IMPAR	$ME_{Schnabel}$	$ME_{Petersen}$	$ME_{CAPTURE}$
cs-en (r)	0.966	0.983	<u>0.992</u>	0.987	0.989	0.988	0.987
de-en (r)	0.859	0.920	0.769	0.934	**0.944**	**<u>0.953</u>**	**<u>0.953</u>**
ru-en (r)	0.868	0.921	<u>0.933</u>	0.896	0.902	0.908	0.908
tr-en (r)	0.938	0.931	0.935	0.959	**<u>0.970</u>**	**0.960**	**0.959**
zh-en (r)	0.894	0.943	0.889	0.933	**<u>0.957</u>**	0.936	0.936
Average (r)	0.905	0.940	0.904	0.942	**<u>0.952</u>**	**0.949**	**0.949**

Table 2: Pearson correlations on the WMT17 metrics task. The best correlation of each language pair is underlined, while correlations of our methods that outperform or match any other method are in bold.

Metrics	BERTSCORE	MOVERSCORE	PRD	IMPAR	$ME_{Schnabel}$	$ME_{Petersen}$	$ME_{CAPTURE}$
Responsiveness (k)	-	0.482	0.398	0.481	**0.483**	**<u>0.487</u>**	**0.484**
Responsiveness (r)	0.739	<u>0.754</u>	0.564	0.743	0.739	0.683	0.747
Responsiveness (p)	0.580	0.594	0.501	0.594	**0.595**	**<u>0.598</u>**	**0.596**
Pyramid (k)	-	0.550	0.444	0.541	0.548	**0.555**	**<u>0.565</u>**
Pyramid (r)	0.823	<u>0.831</u>	0.658	0.804	0.813	0.770	0.808
Pyramid (p)	0.703	0.701	0.588	0.693	0.698	**0.704**	**<u>0.718</u>**

Table 3: Summary-level correlations to human evaluation on TAC 2009. Bold values represent correlations from our proposed metrics that outperform or match all the other methods. The best correlations of each score are underlined. BERTScore results were taken from Zhao et al. (2019).

6.2 Text summarization

We further assessed text summarization with the TAC-2009 dataset[3], consisting of news articles from ten different topics, with four reference summaries and fifty-five evaluation summaries from summarization systems per article. We evaluate each evaluation summary independently, performing a summary-level evaluation. Two scores were assigned to each evaluation summary: the *pyramid score*, which evaluates the semantic similarity between the reference and evaluation summaries, and the *responsiveness score*, that measures the overall quality of the evaluation summary in terms of grammar and content.

Table 3 shows the Kendall (k), Pearson (r), and Spearman (p) correlation to human evaluation for each score. Considering Kendall and Spearman correlations, our family of methods outperforms all the rest on responsiveness score. Moreover, $ME_{Petersen}$ and $ME_{CAPTURE}$ outperform all methods on the above correlations on the pyramid score. Considering Pearson correlation, our family of methods outperforms PRD, and at least one of our metrics consistently outperforms IMPAR on both scores. We hypothesize that the lower Pearson correlations of our metrics could be explained by the instability of the population estimation process due to the low amount of samples, *i.e.* reference and evaluation words.

7 Conclusion

In this work, we present a family of methods derived from popular population size estimators that have been widely used in ecology in the past several decades. We show that our family of methods is able to assess language systems under different representations effectively, *i.e.* using contextualized word and sentence embeddings. Our methods show a high correlation to human evaluation on challenging language generation tasks as well as the desired sensitivity to detect mode collapse and quality detriment.

In the future, we would like to evaluate our family of metrics on image generation tasks, reinforcing the general applicability of our methods. Moreover, we plan to extend our family of methods to also cover popular open populations estimation methods, where the population size may vary over time. In the end, we hope that combining the information from closed and open population methods will improve the overall assessment of language systems, further fostering the adoption of ecology methods in NLP.

---

[3] http://tac.nist.gov/

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Ondřej Bojar, Yvette Graham, and Amir Kamran. 2017. Results of the WMT17 metrics shared task. In *Proceedings of the Second Conference on Machine Translation*, pages 489–513, Copenhagen, Denmark, September. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, September. Association for Computational Linguistics.

Ondřej Cífka, Aliaksei Severyn, Enrique Alfonseca, and Katja Filippova. 2018. Eval all, trust a few, do wrong to none: Comparing sentence generation models. *arXiv preprint arXiv:1804.07972*.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark, September. Association for Computational Linguistics.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Saadia Gabriel, Antoine Bosselut, Ari Holtzman, Kyle Lo, Asli Celikyilmaz, and Yejin Choi. 2019. Cooperative generator-discriminator networks for abstractive summarization with narrative flow. *arXiv preprint arXiv:1907.01272*.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6626–6637. Curran Associates, Inc.

Valentin Khrulkov and Ivan Oseledets. 2018. Geometry score: A method for comparing generative adversarial networks. *arXiv preprint arXiv:1802.02664*.

Charles J Krebs et al. 1989. Ecological methodology. Technical report, Harper & Row New York.

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. 2019. Improved precision and recall metric for assessing generative models. In *Advances in Neural Information Processing Systems 32*, pages 3927–3936. Curran Associates, Inc.

Siyao Li, Deren Lei, Pengda Qin, and William Yang Wang. 2019. Deep reinforcement learning with distributional semantic rewards for abstractive summarization. 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 6038–6044, Hong Kong, China, November. Association for Computational Linguistics.

Zinan Lin, Ashish Khetan, Giulia Fanti, and Sewoong Oh. 2018. Pacgan: The power of two samples in generative adversarial networks. In *Advances in neural information processing systems*, pages 1498–1507.

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

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019a. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019b. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota, June. Association for Computational Linguistics.

James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2019. Putting evaluation in context: Contextual embeddings improve machine translation evaluation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2799–2808, Florence, Italy, July. Association for Computational Linguistics.

Gonçalo Mordido, Haojin Yang, and Christoph Meinel. 2018. Dropout-gan: Learning from a dynamic ensemble of discriminators. *arXiv preprint arXiv:1807.11346*.

G. Mordido, H. Yang, and C. Meinel. 2020. microbatchgan: Stimulating diversity with multi-adversarial discrimination. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3050–3059.

Courtney Napoles, Matthew R 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.

Julian Niedermeier, Gonçalo Mordido, and Christoph Meinel. 2020. Improving the evaluation of generative models with fuzzy logic. *arXiv preprint arXiv:2002.03772*.

David L Otis, Kenneth P Burnham, Gary C White, and David R Anderson. 1978. Statistical inference from capture data on closed animal populations. *Wildlife monographs*, pages 3–135.

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, July. 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, November. Association for Computational Linguistics.

William Edwin Ricker. 1975. Computation and interpretation of biological statistics of fish populations. *Bull. Fish. Res. Bd. Can.*, 191:1–382.

Mehdi S. M. Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. 2018. Assessing generative models via precision and recall. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 5228–5237. Curran Associates, Inc.

Jonathan Sauder, Ting Hu, Xiaoyin Che, Goncalo Mordido, Haojin Yang, and Christoph Meinel. 2020. Best student forcing: A simple training mechanism in adversarial language generation. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4680–4688.

Zoe Emily Schnabel. 1938. The estimation of the total fish population of a lake. *The American Mathematical Monthly*, 45(6):348–352.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (to appear)*. Association for Computational Linguistics.

Stanislau Semeniuta, Aliaksei Severyn, and Sylvain Gelly. 2018. On accurate evaluation of gans for language generation. *arXiv preprint arXiv:1806.04936*.

Akash Srivastava, Lazar Valkov, Chris Russell, Michael U Gutmann, and Charles Sutton. 2017. Veegan: Reducing mode collapse in gans using implicit variational learning. In *Advances in Neural Information Processing Systems*, pages 3308–3318.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.

Ryoma Yoshimura, Hiroki Shimanaka, Yukio Matsumura, Hayahide Yamagishi, and Mamoru Komachi. 2019. Filtering pseudo-references by paraphrasing for automatic evaluation of machine translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 521–525, Florence, Italy, August. Association for Computational Linguistics.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Junbo Zhao, Yoon Kim, Kelly Zhang, Alexander Rush, and Yann LeCun. 2018. Adversarially regularized autoencoders. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5902–5911, Stockholmsmässan, Stockholm Sweden, 10–15 Jul. PMLR.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. 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 563–578, Hong Kong, China, November. Association for Computational Linguistics.

A Theoretical discussions

We will briefly study the validity of our methods when assessing two equal sets, *i.e.* when the reference set is identical to the evaluation set. Formally, we define that:

Definition A.1. Two sets S and S' are equal if $S \subseteq S'$ and $S' \subseteq S$.

Theorem A.1. *Considering two equal sets S and S', $ME_{Petersen}(S, S')$ returns its maximum score of 1.*

Proof. Since $\sum_{s' \in S'} f(s', S) = \sum_{s \in S} f(s, S') = |S|$, following Equation 3 we have:

$$\widehat{P}_{\text{Petersen}}(S, S') = \frac{2|S| \times 2|S|}{2|S|} = 2|S|. \tag{9}$$

Since $P = 2|S|$, following Equation 1 we have:

$$A(P, \widehat{P}_{\text{Petersen}}(S, S')) = A(2|S|, 2|S|) = 0. \tag{10}$$

Finally, adopting Equation 8, we conclude the proof:

$$\text{ME}_{\text{Petersen}}(S, S') = 1 - A(P, \widehat{P}_{\text{Petersen}}(S, S')) = 1 - 0 = 1. \tag{11}$$

$\square$

Theorem A.2. *Considering two equal sets S and S', $ME_{Schnabel}(S, S')$ returns its maximum score of 1.*

Proof. Since $\sum_{s' \in S'} \sum_{s \in S} f(s, \text{NN}_k(s', S')) = |S| \times (K + 1)$ and $\sum_{i=1}^{|S'|} \sum_{j=1}^{|S|} \Big(f(s_j, \text{NN}_k(s'_i, S')) + |M(i, S, S') \cap \text{NN}_k(s'_i, S')| \Big) = |S'| \times \Big((K+1) + (K+1) \Big)$, following Equation 4 we have:

$$\widehat{P}_{\text{Schnabel}}(S, S') = \frac{\Big((K+1) \times |S'| + |S| \times (K+1) \Big) \times 2|S|}{|S'| \times \Big((K+1) + (K+1) \Big)} = \frac{\Big(2 \times |S| \times (K+1) \Big) \times 2|S|}{\Big(2 \times |S| \times (K+1) \Big)} = 2|S|. \tag{12}$$

Since $P = 2|S|$, following Equation 1 we have:

$$A(P, \widehat{P}_{\text{Schnabel}}(S, S')) = A(2|S|, 2|S|) = 0. \tag{13}$$

Finally, adopting Equation 8, we conclude the proof:

$$\text{ME}_{\text{Schnabel}}(S, S') = 1 - A(P, \widehat{P}_{\text{Schnabel}}(S, S')) = 1 - 0 = 1. \tag{14}$$

$\square$

Theorem A.3. *Considering two equal sets S and S', $ME_{CAPTURE}(S, S')$ returns its maximum score of 1.*

Proof. Since $\sum_{s \in S} \sum_{s' \in S'} \Big(f(s', \text{NN}_k(s, S)) + |\text{NN}_k(s, S)| \Big) = \sum_{s' \in S'} \sum_{s \in S} \Big(f(s, \text{NN}_k(s', S')) + |\text{NN}_k(s', S')| \Big) = 2|S|(K+1)$, following Equation 6 we have:

$$L_n(P_{\text{CAPTURE}}; S, S') = \ln \Big(\frac{P_{\text{CAPTURE}}!}{(P_{\text{CAPTURE}} - 2|S|)!} \Big) + 4|S|(K+1) \times ln(4|S|(K+1)) + (2|S|$$

$$P_{\text{CAPTURE}} - 4|S|(K+1)) \times \ln(2|S|P_{\text{CAPTURE}} - 4|S|(K+1)) - (2|S|P_{\text{CAPTURE}}) \ln(2|S|P_{\text{CAPTURE}}). \tag{15}$$

Iterating through $P_{\text{CAPTURE}} \in \mathbb{N}_{\geq 2|S|}$ and substituting $|S| = 10$ and $K = 1$ as an illustration, we have:

$$L_N(2|S|; S, S') \approx -158 \quad and \quad L_N(2|S| + 1; S, S') \approx -159. \tag{16}$$

Since $L_n(2|S|; S, S') > L_n(2|S| + 1; S, S')$, $P_{\text{CAPTURE}} = 2|S|$ maximizes the likelihood function and, following Equation 7, $\widehat{P}_{\text{CAPTURE}} = 2|S|$.

Since $P = 2|S|$, following Equation 1 we have:

$$A(P, \widehat{P}_{\text{CAPTURE}}(S, S')) = A(2|S|, 2|S|) = 0. \tag{17}$$

Finally, adopting Equation 8, we conclude the proof:

$$\text{ME}_{\text{CAPTURE}}(S, S') = 1 - A(P, \widehat{P}_{\text{CAPTURE}}(S, S')) = 1 - 0 = 1. \tag{18}$$

$\square$

B Additional experiments on dialogue generation

Human evaluation correlation on assessing language generation with default K, as well as a comparison with InferSent and SBERT embeddings from BERT-base and BERT-large, are provided in Table 4. We observe that the relative performance between all methods does not change when compared to using the best K, with our family of methods showing the overall best performance between the compared methods and embeddings. Furthermore, SBERT-based embeddings tend to show a higher correlation than InferSent embeddings across all correlations and methods, with the exception of IMPAR's r and k. This goes in accordance with several recent works that show that contextualized embeddings from BERT seem to help across a wide variety of tasks (Liu et al., 2019b; Li et al., 2019; Gabriel et al., 2019; Mathur et al., 2019; Yoshimura et al., 2019).

C Additional experiments on machine translation

We further experimented with assessing machine translation systems using contextualized sentence embeddings. To achieve this, we use all the reference translations as reference samples and the translations of each translation system as evaluation samples. We perform this assessment individually for each translation system available for each language pair.

Pearson (r) correlations are presented in Table 5. Considering the average across all language pairs, our family of methods outperforms PRD and IMPAR. Note that, as expected, using contextualized sentence embeddings shows lower performance than contextualized word embeddings (Table 6) due to the finer-granularity of the assessment in the latter case.

CORR.	FID			PRD			IMPAR			ME$_{\text{Schnabel}}$			ME$_{\text{Petersen}}$			ME$_{\text{CAPTURE}}$		
	ISENT	sSBERT	SBERT	ISENT	sSBERT	SBERT	ISENT	sSBERT	SBERT	ISENT	sSBERT	SBERT	ISENT	sSBERT	SBERT	ISENT	sSBERT	SBERT
r	0.838	0.860	0.902	0.669	0.661	0.684	0.708	0.629	0.633	**0.905**	**0.883**	**0.917**	0.824	0.828	0.872	**0.903**	**0.882**	**0.902**
k	0.511	0.867	0.867	0.689	0.778	0.822	0.689	0.467	0.600	**0.778**	**<u>0.911</u>**	**<u>0.911</u>**	**<u>0.867</u>**	**<u>0.911</u>**	**<u>0.911</u>**	**0.867**	**<u>0.911</u>**	**0.867**
p	0.733	0.964	0.964	0.830	0.903	0.939	0.879	0.552	0.903	**0.903**	**<u>0.976</u>**	**<u>0.976</u>**	**<u>0.964</u>**	**<u>0.976</u>**	**<u>0.976</u>**	**<u>0.964</u>**	**<u>0.976</u>**	**0.964**

Table 4: Correlations to human evaluation regarding the fluency of 10 different models with default K. ISENT refers to InferSent embeddings, sSBERT refers to sentence embeddings from BERT-base ('bert-base-nli-mean-tokens'), and SBERT refers to sentence embeddings from BERT-large ('bert-large-nli-mean-tokens'). For each embedding type, best scores of each correlation are underlined, while bold values represent the correlations where our methods outperform or match all of the other methods' performance. Absolute correlation values are presented for FID.

Translations	PRD	IMPAR	ME$_{\text{Schnabel}}$	ME$_{\text{Petersen}}$	ME$_{\text{CAPTURE}}$
cs-en (r)	0.979	0.912	**<u>0.993</u>**	**0.989**	0.971
de-en (r)	<u>0.885</u>	0.794	0.869	0.857	0.845
ru-en (r)	0.931	0.897	**<u>0.949</u>**	0.904	0.842
tr-en (r)	0.833	0.866	0.857	**0.874**	**<u>0.922</u>**
zh-en (r)	0.582	0.727	**0.745**	**0.735**	**<u>0.839</u>**
Average (r)	0.842	0.839	**<u>0.883</u>**	**0.872**	**0.850**

Table 5: Pearson correlations for the WMT17 metrics task using contextualized sentence SBERT embeddings (*'bert-base-nli-max-tokens'*) with default K. The best correlation of each language pair is underlined. Correlations where our methods outperform or match all of the other methods are highlighted in bold.

Translations	PRD	IMPAR	ME$_{\text{Schnabel}}$	ME$_{\text{Petersen}}$	ME$_{\text{CAPTURE}}$
cs-en (r)	<u>0.992</u>	0.981	0.975	0.970	0.971
de-en (r)	0.769	0.934	**0.940**	**<u>0.953</u>**	**<u>0.953</u>**
ru-en (r)	<u>0.933</u>	0.888	0.887	0.895	0.897
tr-en (r)	0.935	<u>0.941</u>	0.917	0.917	0.929
zh-en (r)	0.889	0.895	**<u>0.957</u>**	**0.936**	**0.936**
Average (r)	0.904	0.928	**0.935**	**0.934**	**<u>0.937</u>**

Table 6: Pearson correlations for the WMT17 metrics task using contextualized word embeddings with default K. The best correlation of each language pair is underlined. Correlations where our methods outperform or match all of the other methods are highlighted in bold.