domain-specific features that aid in text coherence evaluation. More specifically, we show it can be leveraged for Human-AI discrimination and used to compare generation qualities of different models with respect to coherence. From the LLM detection task results, we find when training with an inclusive corpus, $\hat{\sigma}_m$ can potentially be utilized to identify sub-domains. Our results also demonstrate that while Entity Grid is a simple and effective measure in artificial settings, it falters in downstream tasks.

While our approach has yielded positive results, we also identify several limitations. In this paper, our focus has been on a single dataset to showcase the usability of BBScore. Although we have demonstrated its ability to discern different LLM-generated content by training with a cross-domain corpus, the obtained results are not entirely satisfactory and still require further robustness validation. In light of this, we acknowledge that the assumption we have made, linking σ_m^2 to specific domain/style, is rather restrictive in its general applicability. Future endeavors will involve expanding the parameter space by introducing higher-dimensional variance estimates. Furthermore, given BBScore's demonstrated capacity to differentiate between LLM-generated text and human-authored text, we aspire to establish BBScore as a more general and well-defined metric for comparing various LLMs in our forthcoming research. Lastly, BBScore's correlations with formal Human evaluation should be examined.

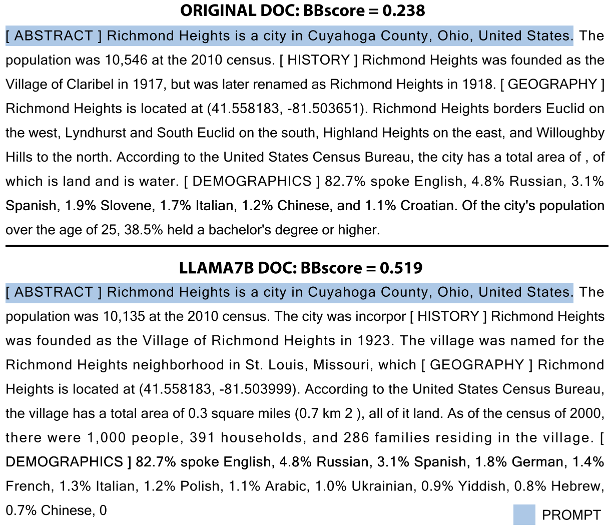
8 Conclusion

Overall, BBScore presents a novel perspective on text coherence and has demonstrated its efficacy on artificial tasks involving deliberately induced incoherence. Additionally, we illustrate the practical utility of BBScore in a natural context, where unintentional deviations from desired coherent text occur. Serving as an intermediate computed score, BBScore holds the potential to become a valuable feature in numerous real-world applications, including tasks related to Human-AI discrimination. In contrast to the intricate network architectures employed in neural entity-grid models (e.g., Multi-layer LSTM), our approach utilizes a simple three-layer perceptron with BBScore as input for classification tasks, devoid of any crafted loss function. Remarkably, the experiments show this approach attains comparable, and in some cases, even superior results. Further refinement of our existing method promises to be an intriguing avenue for future research.

Appendix

A Example of pairwise discrimination on AI discrimination tasks

In the pairwise discrimination on AI discrimination tasks, for each data pair (original doc, AI-generated doc), we will compute the BBScore separately as we did in the coherence task, the local/global discrimination task. An example is shown in Figure S2, here the AI-generated document uses the first sentence of the original doc as a prompt and generates the rest document with LLaMA7b. In this example, the original document has a lower BBScore and the task on this data pair is marked as successful.



ORIGINAL DOC: BBscore = 0.238

[ABSTRACT] Richmond Heights is a city in Cuyahoga County, Ohio, United States. The population was 10,546 at the 2010 census. [HISTORY] Richmond Heights was founded as the Village of Claribel in 1917, but was later renamed as Richmond Heights in 1918. [GEOGRAPHY] Richmond Heights is located at (41.558183, -81.503651). Richmond Heights borders Euclid on the west, Lyndhurst and South Euclid on the south, Highland Heights on the east, and Willoughby Hills to the north. According to the United States Census Bureau, the city has a total area of , of which is land and is water. [DEMOGRAPHICS] 82.7% spoke English, 4.8% Russian, 3.1% Spanish, 1.9% Slovene, 1.7% Italian, 1.2% Chinese, and 1.1% Croatian. Of the city's population over the age of 25, 38.5% held a bachelor's degree or higher.

LLAMA7B DOC: BBscore = 0.519

[ABSTRACT] Richmond Heights is a city in Cuyahoga County, Ohio, United States. The population was 10,135 at the 2010 census. The city was incorpor [HISTORY] Richmond Heights was founded as the Village of Richmond Heights in 1923. The village was named for the Richmond Heights neighborhood in St. Louis, Missouri, which [GEOGRAPHY] Richmond Heights is located at (41.558183, -81.503999). According to the United States Census Bureau, the village has a total area of 0.3 square miles (0.7 km 2), all of it land. As of the census of 2000, there were 1,000 people, 391 households, and 286 families residing in the village. [DEMOGRAPHICS] 82.7% spoke English, 4.8% Russian, 3.1% Spanish, 1.8% German, 1.4% French, 1.3% Italian, 1.2% Polish, 1.1% Arabic, 1.0% Ukrainian, 0.9% Yiddish, 0.8% Hebrew, 0.7% Chinese, 0

Figure 8: An example of the pairwise AI discrimination

B Domain generalizability of Brownian encoders

We use the WikiSection encoder to encode GCDC texts and obtain the corresponding BBscores. A three-layer perceptron is then trained on the BBscores for a three-class classification task on the GCDC dataset. The results are shown in Table S1.

Dataset	**Domain**			
	Enron	Clinton	Yahoo	Yelp
Train	47.67	43.11	49.54	51.45
Test	47.50	41.50	42.64	49.25

Table 6: Three-classs Classification Task Results on GCDC Dataset with the WikiSection Encoder.

C Diffusion coefficient $\hat{\sigma}_m^2$ analysis

As shown in Figure S1, it describes the AUC score for the blocksize=1 shuffle test under different diffusion coefficients. It shows our current approximation of the diffusion coefficient (marked by the red dashed line) can give us a better result but not the best (marked by the olive dashed line). Moreover, it also shows, the standard Brownian bridge σ_m^2=1 shows a poor result AUC score=1 which emphasizes the necessity of this diffusion coefficient approximation.

D BBScore defined with a shifting window

The basic BBScore is defined as:

$$B(\mathbf{s}|\hat{\sigma}_m^2) = \frac{|\sum_{i=2}^{T(\mathbf{s})-1} \ln(\alpha_i(\mathbf{s})\hat{\sigma}_m^2) + \frac{\beta_i(\mathbf{s})}{\hat{\sigma}_m^2}|}{T(\mathbf{s}) - 2}. \tag{3}$$

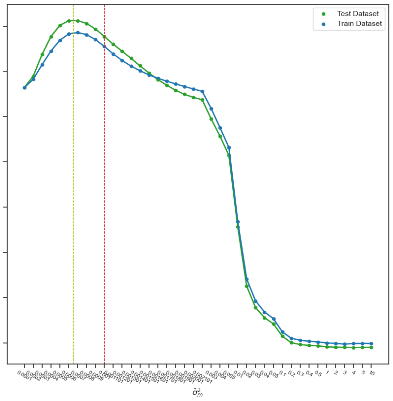


Figure 9: Diffusion coefficient analysis: The AUC score for shuffle test with block 1 under different σ_m^2, and the red dash line corresponds to the $\hat{\sigma}_m^2$ approximated with the train dataset.

We also test BBScore with a shifting window to capture the local coherence property: given a shifting window size $2w+1$, $w \in \mathbb{N}$, the shifting window BBScore $B_w(\mathbf{s}|\hat{\sigma}_m^2)$ is defined as,

$$B_w(\mathbf{s}|\hat{\sigma}_m^2) = \frac{|\sum_{i=w+1}^{T(\mathbf{s})-w} \ln(\alpha_{i,w}(\mathbf{s})\hat{\sigma}_m^2) + \frac{1}{\hat{\sigma}_m^2}\beta_{i,w}(\mathbf{s})|}{T(\mathbf{s}) - 2w} \tag{4}$$

where for $i = w+1, \cdots T(\mathbf{s}) - w$

$$\alpha_{i,w}(\mathbf{s}) = 2\pi \frac{w(w+1)}{2w+1},\ \beta_{i,w}(\mathbf{s}) = \frac{(2w+1)||s_i - \mu_i||^2}{2w(w+1)}$$

and

$$\mu_i = s_{i-w} + \frac{w+1}{2w+1}(s_{i+w} - s_{i-w}).$$

E Training details

For all the experiments mentioned in the main paper, we used a hidden dimension size of 128 for the multi-layer perceptron appended to the GPT2 encoder and an output dimension size of 8 for the fully connected layer. The Brownian encoders were then trained using the contrastive objective function via the SGD optimizer, with learning rate of 1×10^{-4}, momentum of 0.9, and batch size of 32. The GPT2-based encoder was trained on 1 node with 2 A100 GPUs and 32 GB of memory.

suscipit ratione dolorem assumenda voluptatem eos id, consequuntur quasi a debitis dolores officia hic laboriosam eligendi molestias?Possimus laudantium est qui libero a quos natus minima quis, voluptate nulla voluptatibus vitae ad tenetur impedit nisi debitis deleniti reiciendis excepturi, iusto velit voluptatem nemo cum reiciendis mollitia?Inventore nobis quis ipsa praesentium doloribus at, consectetur reprehenderit suscipit ipsa molestias quisquam ipsam adipisci itaque odit error, architecto nisi laborum dolorum quos dolore officia fuga quisquam, et eveniet debitis doloribus sint.Sint saepe corrupti doloremque et deserunt placeat ut cum amet eos, inventore officiis dolor suscipit aperiam tempora ea, nulla ipsa et cumque qui ducimus nostrum earum, esse magni harum tenetur aspernatur?Asperiores vel quo repudiandae voluptate sapiente, illo consequatur impedit quo voluptas quam reprehenderit blanditiis temporibus perspiciatis sint, obcaecati placeat aliquid distinctio cum numquam ad sequi enim expedita ipsam.Nemo incidunt tenetur quaerat et quasi qui iure veritatis aspernatur cumque, repellendus placeat accusantium nam voluptatibus alias eligendi voluptatum, itaque voluptatem quod nam ex cum iste.Delectus unde magnam assumenda provident praesentium commodi, cumque sit dignissimos illo iure amet odio voluptatibus nobis ipsam officiis, unde sequi molestiae, ad eligendi cumque minima voluptatum dolorem adipisci distinctio quasi ipsum sequi.Aperiam doloribus tempora repellendus provident ratione, vero itaque cum deserunt velit labore, iusto ipsam neque minima minus consectetur.Magni provident placeat non veniam itaque perspiciatis odit dolor, provident excepturi fuga voluptatem animi tempore, et earum laborum provident necessitatibus nisi quisquam vel vero, mollitia similique nulla itaque nisi libero doloremque iure delectus perferendis, ducimus corrupti possimus consequuntur odio vel.Cupiditate dicta officiis repellendus eligendi dignissimos consequatur in et nostrum maxime quo, est veritatis possimus quisquam at hic nihil in quae ut reiciendis, sequi dignissimos exercitationem necessitatibus et ea.In temporibus dignissimos impedit earum, illo eveniet omnis, expedita veritatis mollitia a libero delectus aliquid similique dicta nisi magni, optio porro laudantium ex libero tempore dolor quaerat amet voluptatibus odio, quia dolorem fugit quae obcaecati ullam laborum.Nisi blanditiis recusandae deserunt repudiandae doloribus reprehenderit, dolore commodi consequuntur sapiente assumenda porro ipsa ipsum, consectetur quo doloremque facere optio mollitia aut quasi sunt iusto.At alias consequuntur, sunt fuga maiores, voluptas aliquam provident molestias soluta, sapiente delectus consectetur nihil ipsum a nam nesciunt?Voluptatibus qui vero, ullam nihil quia eum laborum adipisci veritatis voluptate voluptates ea et odit, omnis aliquam nam porro hic praesentium sequi nihil nesciunt quibusdam ratione at, vitae eligendi voluptatibus voluptate quidem numquam ratione laborum accusantium nisi maiores distinctio.Beatae in atque accusamus reprehenderit, distinctio culpa deleniti recusandae repellat dolore assumenda mollitia, blanditiis sint sunt veritatis similique ducimus maxime, cumque libero et eum dicta?Illo minima blanditiis voluptates totam, cupiditate facere fugit quidem quasi culpa deserunt at expedita mollitia eveniet, ducimus dolore fugiat enim molestias totam consectetur repellat odit, et illum dolor quia earum magni facere iusto, quasi consequuntur perspiciatis?Praesentium mollitia id nobis ad eveniet debitis expedita excepturi facere deleniti, voluptate tenetur obcaecati, aspernatur nihil eius, praesentium qui corporis nulla vel error pariatur.Harum tempora assumenda maxime sed beatae delectus perferendis facere praesentium, quas fuga error earum provident iusto aliquam repellendus sapiente, quod eveniet expedita deleniti vel recusandae fuga quidem temporibus, nulla autem eum omnis tenetur alias a aut illo incidunt esse ea.Sequi sapiente reprehenderit porro modi dicta laborum quibusdam odio similique pariatur reiciendis, eaque consequuntur corrupti?Quam autem illum nam eum sapiente aperiam saepe molestias dignissimos non obcaecati, iste vero neque tempora exercitationem quasi perferendis commodi officia fuga, quaerat rem sequi velit natus consequuntur at.Deleniti placeat et ullam suscipit ea asperiores repellat odit, tenetur aut error, nostrum voluptatum voluptas quae voluptates deserunt optio magnam, soluta totam molestiae consequatur voluptas, quo nemo nam fugiat quam.