AutoESD: An Automated System for Detecting Non-authentic Texts for High-Stakes Writing Tests

Ikkyu Choi, Jiangang Hao, Chen Li, Michael Fauss & Jakub Novák

Abstract

A frequently encountered security issue in writing tests is non-authentic text submission: test takers submit texts that are not of their own but are copies of texts prepared by someone else. In this paper, we propose AutoESD, a human-in-the-loop and automated system to detect non-authentic texts for a large-scale writing test, and report its performance on an operational data set. The AutoESD system utilizes multiple automated text similarity measures to identify suspect texts and provides an analytics-enhanced web application to help human expert to review the identified texts. To evaluate the performance of AutoESD, we obtained its similarity measures on TOEFL iBT writing responses collected from multiple remote administrations and examined their distributions. The results were highly encouraging in that the distributional characteristics of AutoESD similarity measures were effective in identifying suspect texts and that the measures could be computed quickly without affecting the operational score turnaround timeline at all.

1 Introduction

A frequently encountered security issue in writing tests is non-authentic text submission: test takers submit texts that are not of their own but are copies of texts prepared by someone else. Broadly speaking, there are two approaches to address this issue. The first approach involves preemptively identifying source texts (and those who prepare them) and detecting submitted texts based on the identified sources. This approach is easy to implement if the source identification is feasible.

However, with large-scale, high-stakes tests, there are often many sources, and many people who prepare them, making preemptive source identification immensely challenging, if not downright impossible. This practical challenge motivates the second approach, in which non-authentic texts are identified by comparing all possible pairings of submitted texts. As long as the same source is used more than once, this pairwise comparison approach allows test developers to identify sources and detect non-authentic texts based on them without the need for a priori knowledge.

The main challenge in implementing the pairwise comparison approach is the sheer number of comparisons needed. With N unique texts, there are $\frac{N \times (N-1)}{2}$ unique pairs to compare. If a test has a large enough volume to make preemptive source identification challenging, the number of texts submitted for each test administration is likely to be large as well; it would be impractical to manually review all incoming text pairs. It is in this context that automated evaluation of text similarity (AETS) holds a promise. AETS is typically much faster than manual text review, which is particularly important in accommodating the large number of text pairs from a large-volume test. AETS also guarantees consistent and replicable results, both of which are crucial characteristics in information to guide high-stakes decisions such as score cancellation.

In this paper, we propose an AETS system for large-scale, high-stakes writing tests and report its performance on an operational data set from a language proficiency test. Declaring a submitted text as non-authentic is a high-stakes decision that can lead to score cancellation. Therefore, test programs often mandate careful reviews by content experts for the decision. The proposed AETS system, which we call AutoESD (short for **Auto**mated **E**ssay **S**imilarity **D**etection), is designed to enhance and facilitate the expert reviews by identifying suspect text pairs and presenting reviewers with identified pairs along with supporting information.

The remainder of this paper is organized as follows. We first provide background information by contextualizing the problem and introducing relevant AETS approaches. We then describe the constituents of the AutoESD system in detail, followed by the presentation of our methodology for the real-world data evaluation and its findings. We conclude this paper with discussions about the implications of our findings, limitations, and future steps.

2 Background

2.1 Non-Authentic Text in Writing Test

A clear and concrete definition of a non-authentic text is elusive in the context of a writing test. From a purist's perspective, any text that is not written by test takers themselves may be considered non-authentic. Applying this perspective to a writing test context, however, can be difficult because there can be a wide range of non-authentic texts in terms of extent and seriousness, from using a few stock phrases to copying an entire text that is written by somebody else. Writing test programs often have scoring policies to handle different degrees of non-authenticity. For example, occasional non-authentic texts may be ignored during scoring, whereas extensive use of non-authentic texts may be considered a security issue that can lead to a substantial penalty such as score cancellation.

2.2 Automated Evaluation of Text Similarity

Automated evaluation of text similarity (AETS) relies on quantifiable definitions of how similar a text is to another text. Similarity between texts can be defined in multiple ways, which result in multiple measures that can be used for AETS. In this section, we review a few that are relevant to this project.

AETS measures can be grouped into two categories depending on whether similarity is defined in terms of raw texts or numeric representations of texts. Text-based AETS measures capture the amount of editing operations needed to turn a text into another. These measures are thus collectively called edit distance measures, and different edit distance measures differ in terms of allowed editing operations. For example, the Levenshtein distance measure (Levenshtein, 1965) includes as editing operations deletion, insertion, and substitition. On the other hand, only deletions and insertions are considered under the longest common substring distance (Needleman & Wunsch, 1970). Other measures, such as the Damerau-Levenshtein distance (Damerau, 1964; Levenshtein, 1965), also count as an editing operation the transposition of adjacent characters. The set of editing operations considered in these text-based AETS measures are all defined at the character level. Consequently, text-based AETS measures capture the distance between two texts at the character level. The similarity between the two text is the flip side of their distance; the further away a text is from the other in terms of an edit distance measure, the less similar they are to each other in terms of that measure, and vice versa.

Text-based AETS measures have been employed in several early systems designed to identify similar content. For example, Heintze (1996) used the amount of consecutive characters shared by a pair of texts to measure their similarity in the context of information retrieval and academic plagiarism detection. Similarly, Manber (1994) proposed the sif system that also utilized consecutive characters as anchors to compare files containing texts and/or source codes within a large database.

The other group of AETS measures relies on representing texts as numeric vectors and calculating the similarity between the vectors. The transformation from a text to a vector can take multiple forms with varying levels of complexity. A simple yet popular transformation involves treating a minimally relevant unit of a text as a single dimension of the resulting vector consisting of frequency counts for those units in a given text. In general, the minimally relevant unit can be a character, a word, a series of words, or somewhere in between. For example, let us consider words as the minimally relevant unit. Suppose there are W unique words among R texts. Let the set of the W words be denoted by $\mathbb{W}$ such that its cardinality $|\mathbb{W}|$ is W. Text r, $1 \leq r \leq R$, can then be represented as a vector of length W, whose first element represents the frequency of the first word (first in $\mathbb{W}$) in the text, the second element represents the frequency of the second word (in $\mathbb{W}$), and so on. Because this transformation regards a text as a set of words, it is often called the bag-of-words model.

The bag-of-words model was at the core of SCAM (Shivakumar & Garcia-Molina, 1995) developed to tackle a wide range of copy detection tasks. Specifically, the mechanism evaluates the amount of shared words between digital materials with word-specific weights obtained in a reference corpus such that less frequent words would contribute more heavily towards measuring similarity than highly frequent words. Similarly, Hadjieleftheriou and Srivastava (2010) proposed a set of similarity measures based on the bag-of-words model with word-specific weights as an efficient way to find relevant answers to a given query within a database.

The bag-of-words model ignores the order of words in original texts, which may not be desirable depending on the use case. To incorporate the word order information, the bag-of-words model can be generalized to consider a sequence of n consecutive words as the minimally relevant unit. Such generalized models are called n-gram models. As an illustrative example, consider the following sentence: `Two dogs are running around the playground.` The sentence consists of seven unique words, six unique bigrams (`two dogs`, `dogs are`, and so on), five unique trigrams (`two dogs are`, `dogs are running`, and so on), and so on. Such n-grams retain the original order of words (within the local window defined by n). Models based on n-grams with $n \geq 2$ can thus utilize the order information.

The n-gram model has been used in multiple plagiarism detection systems. For example, Lyon et al. (2001) have developed Ferret, an academic plagiarism detector, with trigrams (i.e., $n = 3$) at its core. Educational Testing Service also has a legacy plagiarism detection system primarily based on shared trigrams (Tetreault & Chodorow, 2010). Researchers have also devised new detection methods by combining the n-gram model with additional information, such as semantic similarity (e.g., Nahnsen et al., 2005), syntactic information (e.g., Uzuner et al., 2005), and structural dependency (e.g., Mozgovoy et al., 2007).

The bag-of-words and n-gram models capture similarity in terms of surface-level, textual overlaps. This focus on textual overlap may be advantageous under certain use cases but may also limit their usefulness in other contexts, especially those that require capturing semantic similarity. For example, consider two synonyms `sofa` and `couch`. They are much more similar in meaning than, say, `bed` and `car` are to each other. However, those four words will be considered as equally distinct under a model that defines similarity as a degree of textual overlap at the word level. Some recent approaches towards text-to-vector transformation attempt to retain semantic relationships between (sub)words in the transformed vector space (e.g., Devlin et al., 2019; Peters et al., 2018). Their transformation functions are much more involved than counting and are typically obtained by fitting a probabilistic model to large corpora. Elements of the resulting vectors are often called embeddings. The embeddings of individual (sub)words can be aggregated (via various methods) to form a text-level embedding vector. It is also possible to obtain text-level embeddings in a more direct manner rather than aggregating (sub)word embeddings. A well-known example of such an approach is latent semantic analysis (Deerwester et al., 1990), in which a matrix of weighted word counts in texts (words in rows and texts in columns) is approximated via low-rank singular value decomposition, and the element-wise product between the resulting singular values and the right singular vector for a given text is taken as the embedding of that text.

The flexibility of word embeddings has allowed researchers to measure text similarity in traditionally more challenging settings. For example, Britt et al. (2004) utilized text embeddings obtained via latent semantic analysis along with string- and pattern-matching techniques to identify multiple issues in student essays including plagiarism and the lack of proper source attribution. Glavaš et al. (2018) used word embeddings to measure text similarity between documents written in different languages. El Moatez Billah Nagoudi et al. (2018) incorporated word-embeddings as part of a two-stage plagiarism detection system. Word embeddings are typically learned through deep learning methodologies involving multiple layers of hidden variables. Wang et al. (2019) is another example of applying a deep learning methodology to the non-authentic response detection task in the context of a speaking test.

Once texts are transformed into numeric vectors, similarity between two such vectors (or distance which is the flip side of similarity) can be quantified in multiple ways. For example, one can turn all non-zero counts in vectors into one and divide the number of shared elements by the number of unique elements between two texts. This "intersection-over-union" similarity measure ranges from 0 (no shared element between the two) to 1 (complete overlap) and is called the Jaccard similarity index. Another popular measure is cosine similarity, which represents the cosine value of the angle between two vectors of the same dimension (two texts in this use case). Cosine similarity thus ranges from -1 (two opposite text vectors) to 1 (two proportional text vectors).

Plagiarism and copy detection is a popular task with a large body of literature. In addition to the methods we reviewed above, numerous others have been proposed, implemented, and evaluated. An extensive review of this literature goes beyond the scope of this paper, and we refer interested readers to surveys such as Alzahrani et al. (2011) and Foltỳnek et al. (2019). We highlight that not all approaches in the literature would be relevant or effective in detecting non-authentic texts in the context of a writing test. Many methods have been developed to detect highly elaborated and refined copying or plagiarism attempts that do not match surface level texts but retain semantic and syntactic structures (e.g., Hussain & Suryani, 2015) or overall ideas (e.g., Gipp et al., 2014). We believe that such attempts are unlikely in the context of writing tests, in which examinees have a time limit and are monitored by proctors.

3 Objectives

The goal of this project was to develop an effective tool to identify and review non-authentic texts in the context of a large-scale writing test. To achieve this goal, we propose AutoESD, a human-in-the-loop AETS system that computes similarity scores between texts, selects a subset for review based on the resulting similarity scores, and presents to expert reviewers the selected subset with the similarity scores as well as other relevant information. AutoESD consists of two main components, one for computing the similarity scores and the other for presenting the information to reviewers such that they can make final decisions. The first component is a behind-the-scene computing engine, and the second one takes the form of a web-based dashboard. The computing engine is designed to run regularly as new tests are administered and generate output files that the web dashboard uses for presentation. Reviewers can review the output files with an analytics-enhanced dashboard, evaluate texts with excess similarity, and decide whether to recommend a text for score cancellation or not.

4 AETS Measures in AutoESD

We focused on AETS measures based on textual overlap, namely edit distance and n-gram models. This decision was motivated by the nature of the detection task at hand. In a typical large-scale writing test, many examinees are assigned to the same prompt, and texts written for the shared prompt are expected to be semantically similar, which could lead to a high noise floor for semantic similarity measures based on embeddings. Moreover, given the timed nature of writing tasks, submitted texts frequently include typos, which are difficult to accommodate using embedding-based measures. On the other hand, it is highly unlikely that two independently written texts share a large amount of textual overlap. AETS measures based on edit distance can accommodate typos with little difficulty. AETS measures based on edit distance and n-gram models also have an additional practical advantage of requiring less computing power than embedding-based measures.

We chose three measures for AutoESD: the token set ratio similarity, trigram cosine similarity, and BLEU score (Papineni et al., 2002). All three selected measures quantify similarity between a given text pair in terms of textual overlap but differ in detail. In the remainder of this section, we present the definition and implementation details of each in a separate subsection. We assume a pool of $R \geq 2$ texts, with an individual text denoted by t with subscripts as needed. The set of unique n-grams in the entire pool will be denoted by $\mathbb{U}^n$, and its cardinality will be denoted by $|\mathbb{U}^n| = U^n$. Similarly, the set of unique n-grams in text i, $1 \leq i \leq R$, will be denoted by $\mathbb{U}_i^n$, and its cardinality by U_i^n. We further assume that the n-gram sets are sorted in the alphabetical order.

4.1 Token Set Ratio

The token set ratio measure captures the amount of textual overlap in terms of the edit distance between a given pair of texts, t_i and t_j, $1 \leq i \leq R$ and $1 \leq j \leq R$. The calculation of this measure involves the following set of preprocessing steps for the compared texts.

First, each of the two texts is transformed into a set of words (unigrams); that is, texts t_i and t_j become sets $\mathbb{U}_i^1$ and $\mathbb{U}_j^1$, respectively. The set transformation reduces multiple occurrences of the same word into one element in the resulting set.

The sets $\mathbb{U}_i^1$ and $\mathbb{U}_j^1$ are further divided into three mutually exclusive subsets: the intersection (denoted by $\mathbb{U}_{i\cap j}^1$), the set containing unigrams that belong to text i but not j (denoted by $\mathbb{U}_{i-j}^1$), and the set containing unigrams that belong to text j but not i (denoted by $\mathbb{U}_{j-i}^1$). The three mutually exclusive sets are then compared in three different ways, yielding three distance measures as the following:

$$
\begin{aligned}
d_1 &= \mathrm{D}\left(T(\mathbb{U}_{i\cap j}^1), \mathrm{C}\left(T(\mathbb{U}_{i\cap j}^1), T(\mathbb{U}_{i-j}^1)\right)\right) \\
d_2 &= \mathrm{D}\left(T(\mathbb{U}_{i\cap j}^1), \mathrm{C}\left(T(\mathbb{U}_{i\cap j}^1), T(\mathbb{U}_{j-i}^1)\right)\right) \\
d_3 &= \mathrm{D}\left(\mathrm{C}\left(T(\mathbb{U}_{i\cap j}^1), T(\mathbb{U}_{i-j}^1)\right), \mathrm{C}\left(T(\mathbb{U}_{i\cap j}^1), T(\mathbb{U}_{j-i}^1)\right)\right)
\end{aligned}
$$

where $\mathrm{D}(a, b)$ stands for the Levenshtein edit distance between texts a and b, $\mathrm{T}(\mathbb{A}^1)$ flattens a unigram set $\mathbb{A}^1$ into a text by concatenating all its elements in order (e.g., for a unigram set $\mathbb{A}^1 = \{$"apple", "orange"$\}$, $T(\mathbb{A}^1) =$ "apple orange"), and $\mathrm{C}(x, y)$ represents the concatenation of two texts x and y.

The three distance measures involve texts of different lengths. To account for the impact of length, the distance measures are then normalized in terms of the number of characters. Let $L(t)$ denote the length of text t in characters. The normalized distance measures are obtained as follows:

$$\begin{aligned} d_1^{\text{norm}} &= \frac{d_1}{L\left(T(\mathbb{U}^1_{i\cap j})\right) + L\left(\mathrm{C}\left(T(\mathbb{U}^1_{i\cap j}), T(\mathbb{U}^1_{i-j})\right)\right)} \\ d_2^{\text{norm}} &= \frac{d_2}{L\left(T(\mathbb{U}^1_{i\cap j})\right) + L\left(\mathrm{C}\left(T(\mathbb{U}^1_{i\cap j}), T(\mathbb{U}^1_{j-i})\right)\right)} \\ d_3^{\text{norm}} &= \frac{d_3}{L\left(\mathrm{C}\left(T(\mathbb{U}^1_{i\cap j}), T(\mathbb{U}^1_{i-j})\right)\right) + L\left(\mathrm{C}\left(T(\mathbb{U}^1_{i\cap j}), T(\mathbb{U}^1_{j-i})\right)\right)} \end{aligned}$$

Lastly, the normalized distance measures are transformed to similarity measures by subtracting them from 1:

$$\begin{aligned} s_1 &= 1 - d_1^{\text{norm}} \\ s_2 &= 1 - d_2^{\text{norm}} \\ s_3 &= 1 - d_3^{\text{norm}} \end{aligned}$$

The maximum among the three similarity measures is the token set ratio similarity measure between texts i and j, denoted by S_{ij}^{tsr}:

$$S_{ij}^{\text{tsr}} = \max(s_1, s_2, s_3).$$

4.2 Trigram Cosine Similarity

The trigram cosine similarity measure quantifies the similarity between two texts in terms of three consecutive word sequences (i.e., trigrams). The trigram cosine similarity measure between a given pair of texts within a given pool can be obtained by the following three steps.

First, all texts in the pool are transformed into a length U^3 vector of trigram counts. Specifically, the transformation maps text t_i, $1 \leq i \leq R$, onto a $U^3 \times 1$ vector $\boldsymbol{\lambda}_i$, whose u-th element, $1 \leq u \leq U^3$, represents the count of trigram u in t_i and is called the term frequency (tf) of that trigram. The resulting R vectors can be stacked to form a $R \times U^3$ matrix, denoted by $\boldsymbol{\Lambda}$, whose rows correspond to texts and columns to trigrams. Formally put,

$$\boldsymbol{\Lambda} = \begin{pmatrix} \boldsymbol{\lambda}_1^T \\ \boldsymbol{\lambda}_2^T \\ \vdots \\ \boldsymbol{\lambda}_J^T \end{pmatrix}$$

This matrix is called the document-term matrix of a text pool. The (i, u)-th element of document-term matrix $\boldsymbol{\Lambda}$, denoted by λ_{iu}, represents the term frequency of trigram u in text i.

The second step involves weighting the trigram tf values in the document-term matrix $\boldsymbol{\Lambda}$ using inverse-document frequency (idf; Sparck Jones, 1972). The idf weight of trigram u, denoted by idf_u, is obtained as follows:

$$\text{idf}_u = \log\left(\frac{1+R}{1+\sum_{r=1}^{R} I(\lambda_{ru} \neq 0)}\right) + 1,$$

where $I(a)$ is the indicator function for condition a (i.e., $I(a) = 1$ if condition a is satisfied and 0 otherwise). As the name indicates, the idf of trigram u is large if it appears in a small number of documents in the pool, and vice versa. Let $\boldsymbol{\Omega}$ denote the $U^3 \times U^3$ diagonal matrix with idf_u being its u-th diagonal element, ω_{uu}. Post-multiplying the document-term matrix $\boldsymbol{\Lambda}$ with the idf weight matrix $\boldsymbol{\Omega}$ amounts to, for every trigram u, weighting its tf value with the corresponding idf value. More formally, we obtain the $R \times U^3$ term frequency-inverse document frequency (tf-idf; Salton & Buckley, 1988) matrix, denoted by $\boldsymbol{\Theta}$ as follows:

$$\boldsymbol{\Theta} = \boldsymbol{\Lambda}\boldsymbol{\Omega}.$$

The (i, u)-th element of the tf-idf matrix $\boldsymbol{\Theta}$, denoted by θ_{iu}, represents the tf-idf value of trigram u in text i.

The final step takes the tf-idf matrix $\boldsymbol{\Theta}$ as input and computes the trigram cosine similarity measure between a given text pair within the pool. Specifically, the trigram cosine similarity measure between texts i and j, denoted by S_{ij}^{tc}, is the cosine similarity between the i-th row and the j-th row of the tf-idf matrix $\mathbf{\Theta}$:

$$S_{ij}^{\text{tc}} = \frac{\sum_{u \in \mathbb{U}^3} \theta_{iu}\theta_{ju}}{\sqrt{\sum_{u \in \mathbb{U}^3} \theta_{iu}^2}\sqrt{\sum_{u \in \mathbb{U}^3} \theta_{ju}^2}},$$

where θ_{iu} is the tf-idf value for trigram u in text i (i.e., the (j,u)-th element of the tf-idf matrix $\mathbf{\Theta}$).

Both the shape (i.e., $R \times U^3$) and elements of a tf-idf matrix are specific to the pool of interest. Therefore, the same pair of texts would in general have different tf-idf weighted vector representations in a different pool, which in turn would lead to different trigram cosine similarity measures.

4.3 Bilingual Evaluation Understudy (BLEU)

The BLEU score (Papineni et al., 2002) characterizes the similarity between a reference corpus and candidate texts by measuring the amount of shared n-grams across multiple n and aggregating them. The reference-candidate distinction reflects its original use case in the context of evaluating machine translation output, in which candidate outputs from a machine translation system are compared to high-quality, reference translations made by human experts. Because human experts tend to produce different yet valid translations of the same source text, it is common in the machine translation context to compare a candidate output to multiple references. This practice is reflected in the use of the term reference "corpus" (to refer to multiple reference translations). By considering n-gram overlaps across multiple n, the BLEU score is designed to quantify the reference-candidate similarity in terms of the choice of words as well as their order.

For a given n, the amount of n-gram overlap between a reference corpus and a candidate text is quantified by modified precision of the candidate in reproducing n-grams in the reference corpus. We first discuss the concept of precision in this context and then introduce the modification.

Let the multiset of n-grams in candidate text i be denoted by $\mathfrak{M}_i^n$. The cardinality of this multiset, denoted by M_i^n, is the number of n-grams in text i and does not necessarily equal U_i^n, which represents the number of *unique* n-grams in text i; they are equal if and only if every n-gram in text i occurs only once. Let the multiset of n-grams that appear in candidate text i *and* the reference corpus is denoted by $\mathfrak{M}^n_{i \cap C}$, with C denoting the reference corpus. The cardinality of this multiset, denoted by $M^n_{i \cap C}$, represents the number of n-grams in text i that also appear in the reference corpus. The precision of candidate text i, denoted by P^n_i, is then obtained as follows:

$$P^n_i = \frac{M^n_{i \cap C}}{M^n_i}.$$

The denominator $M^n_{i \cap C}$ ranges from 0 to M^n_i. The corresponding precision P^n_i is thus a rational number between 0 and 1.

The n-gram precision P^n_i may appear appealing as a similarity measure but has a major caveat that involves (almost) degenerating systems. Papineni et al. (2002) illustrated this caveat with the example of a degenerating system that only outputs a single word "`the`" multiple times. Because the definite article is one of the most frequently used words, it is highly likely to be included in any given reference corpus. The resulting unigram precision is thus 1, despite the apparent failure of the candidate text in conveying any meaning, let alone properly approximating references.

Papineni et al. (2002) addressed this caveat by introducing a modified version of the precision measure. Specifically, they modified the precision measure such that the maximum number a given n-gram can contribute towards the shared n-gram count (i.e., the numerator for the precision measure) is set to equal the maximum number of that n-gram appearing in any of the texts constituting the reference corpus. Suppose there are $Q \geq 1$ texts in the reference corpus. Let the multiset of n-grams in reference text q, $1 \leq q \leq Q$, be denoted by $\mathfrak{M}^n_q$. Furthermore, let the multiplicity of n-gram k ($1 \leq k \leq U^n_i$ for candidate text i; $1 \leq k \leq U^n_q$ for reference text q) in the multiset for text a $\mathfrak{M}^n_a$ be denoted by $m^n_a(k)$. We can then write the denominator of the precision as $M^n_i = \sum_{k=1}^{U^n_i} m^n_i(k)$. Similarly, let the number of the same n-gram k appearing in reference text q be denoted by $m^n_q k$. The contribution of n-gram k in text i, denoted by M^n_{ik}, towards the modified precision is bounded by the maximum multiplicity of that n-gram among Q reference texts:

$$M^n_{ik} = \min\left(m^n_i(k), \max_q(m^n_q(k))\right).$$

The modified n-gram precision of candidate text i, denoted by p^n_i, is then obtained as the following:

$$p_i^n = \frac{\sum_k M_{ik}^n}{M_i^n}.$$

The resulting modified precisions are then aggregated across the values of n as a weighted geometric mean of n-gram modified precisions:

$$\exp\left(\sum_{n=1}^{N} w_n \log p_i^n\right),$$

where w_n is the weight for n-gram.

Another caveat of using precision as a similarity measure is underrepresenting reference texts. This caveat is easy to illustrate with an example. Suppose a candidate sentence of `I like apples`, with the corresponding reference sentence of `I like apples and oranges`. The candidate underrepresents the reference but the precision measure, with or without the modification, is 1. To address this issue, Papineni et al. (2002) introduced a penalty to candidate texts that are too short (called the brevity penalty). The concept of being ”too short” is operationalized as being shorter than the shortest of the reference texts in terms of word count (i.e., the number of unigrams). The number of words in candidate text i equals the cardinality of its unigram multiset M_i^1. Similarly, the number of words in reference text q is denoted by M_q^1. Then the brevity penalty of candidate i, denoted by b_i, is obtained as the following:

$$b_i = \begin{cases} 1, & \text{if } M_i^1 > \min_q(M_q^1) \\ \exp(1 - \frac{\min_q(M_q^1)}{M_i^1}), & \text{otherwise.} \end{cases}$$

Finally, the BLEU score of a candidate text i in reference to corpus C, denoted by S_{iC}^{bleu}, is obtained as the following:

$$S_{iC}^{\text{bleu}} = b_i \times \exp\left(\sum_{n=1}^{N} w_n \log p_i^n\right).$$

The weights, one for each n gram involved, can be specified depending on the use case. The most common form of the BLEU score involves the range of n from 1 to 4 with the uniform weight across the four n-gram similarity values. The use of the geometric mean for aggregation leads to the overall zero score whenever any individual n-gram similarity value is zero.

Unlike the token set ratio measure or the trigram cosine similarity, the BLEU score is an asymmetric measure; swapping candidate and reference texts will in general lead to different BLEU scores. When every incoming text is compared to every other in a pairwise manner, the reference-candidate designation within a pair is arbitrary. We thus do not consider the BLEU score for all pairwise comparisons of incoming texts. However, the BLEU score may prove useful in different scenarios in which there exists a natural reference-candidate distinction. For example, a test program may keep a list of non-authentic texts that were detected in previous administrations and plan to compare all incoming texts against the list.

4.4 Implementation

We developed a custom computing pipeline in `python` to obtain the similarity measures using multiple libraries. Computations for the token set ratio measure was implemented using the `token_set_ratio` function in the `RapidFuzz` library (Bachmann, 2021). Specifically, we used the library's `process` module to batch process the computation for efficiency. For the trigram cosine similarity measure, we used the `TfidfVectorizer` function in the `scikit-learn` library (Pedregosa et al., 2011) to transform texts into a tf-idf weighted matrix. The same library's `cosine_similarity` function was then used to obtain the similarity measure between tf-idf weighted text vectors. Lastly, we wrote a custom function to obtain the BLEU scores to avoid any overhead irrelevant to our specific use case.

5 Evaluation Study

5.1 Context: The TOEFL iBT Test and its Writing Section

The TOEFL iBT test is a widely used academic English proficiency assessments whose main use involves informing and guiding higher-education admission decisions for individual test takers (Educational Testing Service, 2023). The test includes four sections: Reading, Listening, Speaking, and Writing, each of which yields a score on the 0 to 30 scale. The total score is the sum of the four section scores and thus ranges from 0 to 120. The Reading and Listening sections consist of selected-response items. The Speaking and Writing sections, on the other hand, consist of constructed-response items for which test takers are asked to produce speech (the Speaking section) and writing (the Writing section) samples. Our focus in this study was on non-authentic text submission in the Writing section. For convenience, we use the term the TOEFL iBT Writing test to refer to the Writing section of the TOEFL iBT test in the remainder of this paper.

At the time of this study (March 2021), the Writing section includes two tasks. The first task presents test takers with a reading passage about a topic, followed by a lecture (an audio file played only once to test takers without its script shown) about the same topic, and asks test takers to write a summary of the lecture in relation to the passage. Because this task requires test takers to integrate information from the passage and lecture in their writing, it is called the integrated task. The second task, called the independent task, asks test takers to write about their opinion and/or preference about a topic with their rationale. Both integrated and independent tasks require test takers that they use their own language; copying any source materials verbatim is not part of the tasks. To reflect the differences in the design and requirements, each task is associated with its own scoring rubric. Submitted texts are scored by a randomly selected rater in the TOEFL iBT professional rater pool and an automated scoring system (Attali & Burstein, 2006) on the six point scale from 0 to 5.

The TOEFL program introduced remote TOEFL iBT administrations (called the TOEFL iBT Home Edition) in 2020. As has been the case with test center administrations, there is an interest in evaluating and identifying non-authentic responses from remote administrations. Another form of non-authentic texts involved verbatim copies of the lecture script for the integrated task.

5.2 Methods

5.2.1 Data

We examined the performance of the AutoESD similarity measures on a data set from the TOEFL iBT test program. Specifically, the data set consisted of operational responses to both integrated and independent tasks from 15 remote administrations. The administrations differed widely in terms of the number of test takers, which in turn led to a wide range of the number of texts across administrations from 127 to 4590. The total number of texts in the data set was 50639, evenly split between the integrated (25347) and independent (25292) tasks. We divided the data set into two subsets according to the task type and analyzed them separately.

5.2.2 Analysis

The main goal of this study presented an unsupervised problem. At the time of this study, the TOEFL iBT data set did not include any label indicating the authenticity of a text, and it was not feasible to add labels based on expert reviews, because conducting such reviews would be extremely costly. Therefore, to evaluate whether texts with excessive similarity can be identified based on the AutoESD similarity measures, we examined the distributions of the similarity measures. Specifically, we considered as the distributional characteristic of an effective measure the following: a low noise floor with a distinct group of outliers with excessively large values. This characteristic would yield clear separation of outliers, which would in turn facilitate the detection of suspect texts.

We evaluated the similarity measure distributions separately for the integrated and independent tasks. The separate analysis plan reflected the difference between the two tasks in design: as noted earlier, the integrated task included multiple sources of information shared by all test takers assigned to the same item, whereas for the independent task test takers only shared a short prompt (often a single sentence). We thus expected that the overall distributions of similarity measures would differ across the two tasks. For both tasks, we obtained the token set ratio and trigram cosine similarity measures for all possible pairings of texts within each administration. The BLEU score was excluded in this pairwise comparison because of its asymmetric nature. To address the challenge specific to the integrated task involving verbatim copies of lecture scripts, we computed all three measures between every incoming text and its associated lecture script.

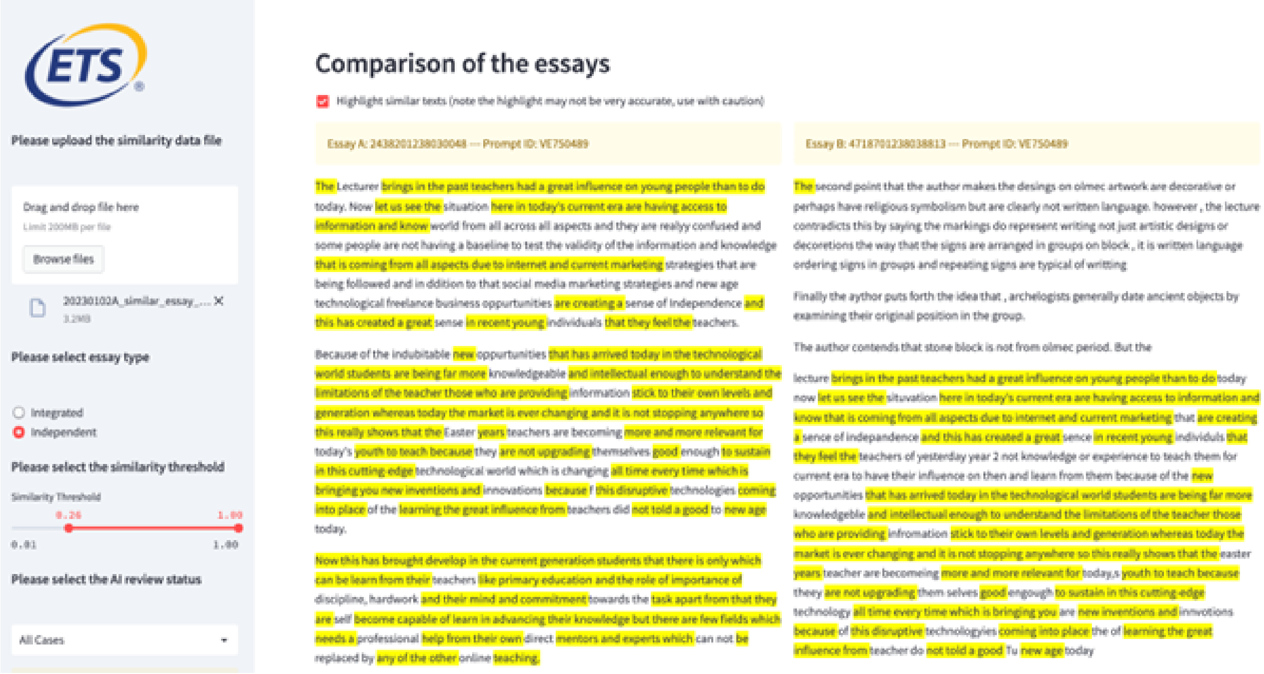
We excluded from all analyses texts that were shorter than 50 words. Test taker responses to TOEFL iBT writing tasks are rarely that short; the total number of excluded texts was 846 (1.7 percent), 577 of which were from the integrated task and the remaining 269 from the independent task. This choice was motivated by multiple factors. First, it is in general difficult to demonstrate proper topic development within such a short text (high-scoring texts tend to have more than 200 words), and therefore, texts that are shorter than 50 words would almost always receive a very low score. Because the main motivation for non-authentic text submission is to receive a high score, it is unlikely for a non-authentic text to be that short. Moreover, such short texts would have led to many unbalanced pairs in terms of length, between which the token set ratio measure is not effective in characterizing similarity. If one text in a pair is much shorter than the other, it's likely that most, if not all, words in the shorter text also appear in the longer text. For an extreme example, when the set of unigrams in text i, $\mathbb{U}_i^1$, is a subset of that in text j, $\mathbb{U}_j^1$, the token set ratio measure between i and j is 1, regardless of how different the two sets are in terms of their size. This characteristic of the token set ratio measure, combined with the fact that non-authentic texts are rarely shorter than 50 words, could have led to many false positives.

We also examined the effectiveness of the measures by means of expert reviews of independent task text pairs that yielded similarity values exceeding thresholds. This review was possible because the number of texts to be reviewed was expected to be much smaller than the entire set of all possible text pairs in the data set. A major challenge in setting up the expert review was the lack of pre-set threshold values for excessive similarity: because this was the first time the AutoESD measures were used in the TOEFL iBT context, we did not know which values to use as thresholds. We thus began with initial threshold values determined based on the distributions of the measures and then iteratively refined them with input from the experts.

5.2.3 Human Expert Review

The integrity and objectivity of automated systems, particularly those driven by artificial intelligence (AI), are under the spotlight these days. As the demand for transparency and fairness in AI escalates, the incorporation of human-centered AI (HCAI) and "human-in-the-loop" methodologies have proven instrumental in mitigating numerous challenges inherent in AI applications. Such human-centric approaches are particularly important when navigating the intricate nuances of determining what constitutes excessive similarity. Determining the threshold beyond which textual overlap can be considered as the result of cheating behaviors requires expert judgment, guided by robust, time-tested policies.

We emphasize that the AutoESD system is not a draconian tool that would lead to automatic score cancellation. Rather, it is a tool designed to work in tandem with human experts, while at the same time assisting experts to make consistent decisions. This design goal for the AutoESD system is represented in its web-based dashboard. A screenshot of the dashboard is presented in Figure 1. The dashboard provided a user interface to review text pairs that were flagged for excessive similarity in terms of the AutoESD measures. Human experts can filter flagged text pairs by the task type (integrated or independent) and similarity values. The dashboard also presented multiple pieces of visual and statistical information, including highlighting of overlapping texts as well as the ratio of overlap within a given text pair.



Please upload the similarity data file

Drag and drop file here
Limit 200MB per file
Browse files

20230102A_similar_essay_... ✕
3.2MB

Please select essay type

○ Integrated
● Independent

Please select the similarity threshold

Similarity Threshold
0.26 1.00
0.01 1.00

Please select the AI review status

All Cases ▾

Comparison of the essays

☑ Highlight similar texts (note the highlight may not be very accurate, use with caution)

Essay A: 2438201238030048 — Prompt ID: VE750489

The Lecturer brings in the past teachers had a great influence on young people than to do today. Now let us see the situation here in today's current era are having access to information and know world from all across all aspects and they are realyy confused and some people are not having a baseline to test the validity of the information and knowledge that is coming from all aspects due to internet and current marketing strategies that are being followed and in didition to that social media marketing strategies and new age technological freelance business oppurtunities are creating a sense of Independence and this has created a great sense in recent young individuals that they feel the teachers.

Because of the indubitable new oppurtunities that has arrived today in the technological world students are being far more knowledgeable and intellectual enough to understand the limitations of the teacher those who are providing information stick to their own levels and generation whereas today the market is ever changing and it is not stopping anywhere so this really shows that the Easter years teachers are becoming more and more relevant for today's youth to teach because they are not upgrading themselves good enough to sustain in this cutting edge technological world which is changing all time every time which is bringing you new inventions and innovations because f this disruptive technologies coming into place of the learning the great influence from teachers did not told a good to new age today.

Now this has brought develop in the current generation students that there is only which can be learn from their teachers like primary education and the role of importance of discipline, hardwork and their mind and commitment towards the task apart from that they are self become capable of learn in advancing their knowledge but there are few fields which needs a professional help from their own direct mentors and experts which can not be replaced by any of the other online teaching.

Essay B: 4718701238038813 — Prompt ID: VE750489

The second point that the author makes the desings on olmec artwork are decorative or perhaps have religious symbolism but are clearly not written language. however , the lecture contradicts this by saying the markings do represent writing not just artistic designs or decoretions the way that the signs are arranged in groups on block , it is written language ordering signs in groups and repeating signs are typical of writting

Finally the aythor puts forth the idea that , archelogists generally date ancient objects by examining their original position in the group.

The author contends that stone block is not from olmec period. But the

lecture brings in the past teachers had a great influence on young people than to do today now let us see the situation here in today's current era are having access to information and know that is coming from all aspects due to internet and current marketing that are creating a sence of independence and this has created a great sence in recent young individuls that they feel the teachers of yesterday year 2 not knowledge or experience to teach them for current era to have their influence on then and learn from them because of the new opportunities that has arrived today in the technological world students are being far more knowledgeble and intellectual enough to understand the limitations of the teacher those who are providing infromation stick to their own levels and generation whereas today the market is ever changing and it is not stopping anywhere so this really shows that the easter years teacher are becomeing more and more relevant for today,s youth to teach because theey are not upgrading them selves good engough to sustain in this cutting edge technology all time every time which is bringing you are new inventions and innvotions because of this disruptive technologyies coming into place the of learning the great influence from teacher do not told a good Tu new age today

Figure 1: Web-based review dashboard enhanced with interactive analytics to facilitate the human expert review.

5.3 Results

5.3.1 All Pairwise Comparisons among Incoming Texts

Figure 2 presents the empirical cumulative distributions of the token set ratio and trigram cosine similarity measures from the first to 99-th percentiles. The empirical distributions differed across the task types at their right tail. Specifically, the integrated task distributions yielded higher similarity values those from the independent task distributions beyond the 90-th percentile. This difference is expected considering that the integrated task provided test takers with much more texts that can be used in their responses (and thus creating opportunities for benign textual overlaps) than the independent task did.

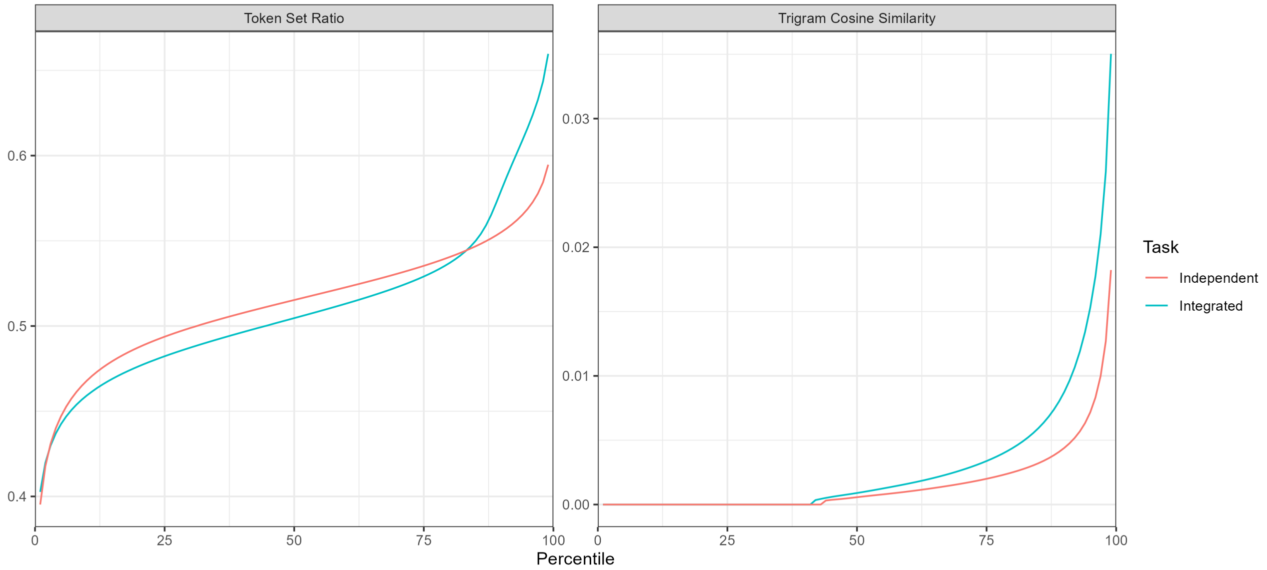


Figure 2: Empirical cumulative distributions of token set ratio and trigram cosine similarity measures from the 1st to 99th percentile values by task type.

Our assumption going into the analysis was that the vast majority of the text pairs would not be based on the same, non-authentic sources. That is, the empirical distributions shown in Figure 2 (i.e., those up to the 99-th percentile values) would constitute noise floors rather than indicating pairs with excess similarity. To examine whether we could interpret these empirical distributions as noise floors, we reviewed a small number of randomly sampled pairs that yielded similarity values near the 99-th percentiles. The review did not reveal any text pairs that were excessively similar to each other, which was in line with our original assumption.

The noise floor of the trigram cosine similarity measure was much lower than the token set ratio measure. Approximately 40 percent of all incoming text pairs yielded zero trigram cosine similarity, whereas the corresponding percentile value for the token set ratio measure was approximately 0.5. This difference arises from the respective definitions of the measures. The token set ratio measure captures word-level overlaps, whereas the trigram cosine similarity is based on overlapping sequences of three consecutive words. It is much more likely for two texts that are not based on the same source to share a common set of words than a common set of trigrams. However, even the noise floor of the token set ratio measure was sufficiently low in that the 99-th percentile value did not go beyond 0.66. There was enough headroom for identifying text pairs with excessively high values for both measures.

Our main interest lay in text pairs with extremely large similarity measures in relation to the overall similarity distributions. Figure 3 shows bivariate distributions of text pairs with similarity measures larger than the 99-th percentile values of the respective cumulative distributions. We observed a qualitative change in the relationship between the two measures starting around the 99.99-th percentile values indicated by the dotted lines in Figure 3. Specifically, the two measures appeared to have only a weak relationship before that point but showed a strong linear relationship after that point. The change in the bivariate relationship was more pronounced for the independent task than for the integrated task. This observation was also supported by correlation coefficient estimates. The Pearson correlation coefficients between the two measures among the pairs below the 99.99th percentiles in the respective distributions were 0.60 and 0.35 for integrated and independent tasks. On the other hand, the corresponding correlation coefficients among the pairs with similarity values larger than the 99.99th percentiles were 0.79 and 0.91.

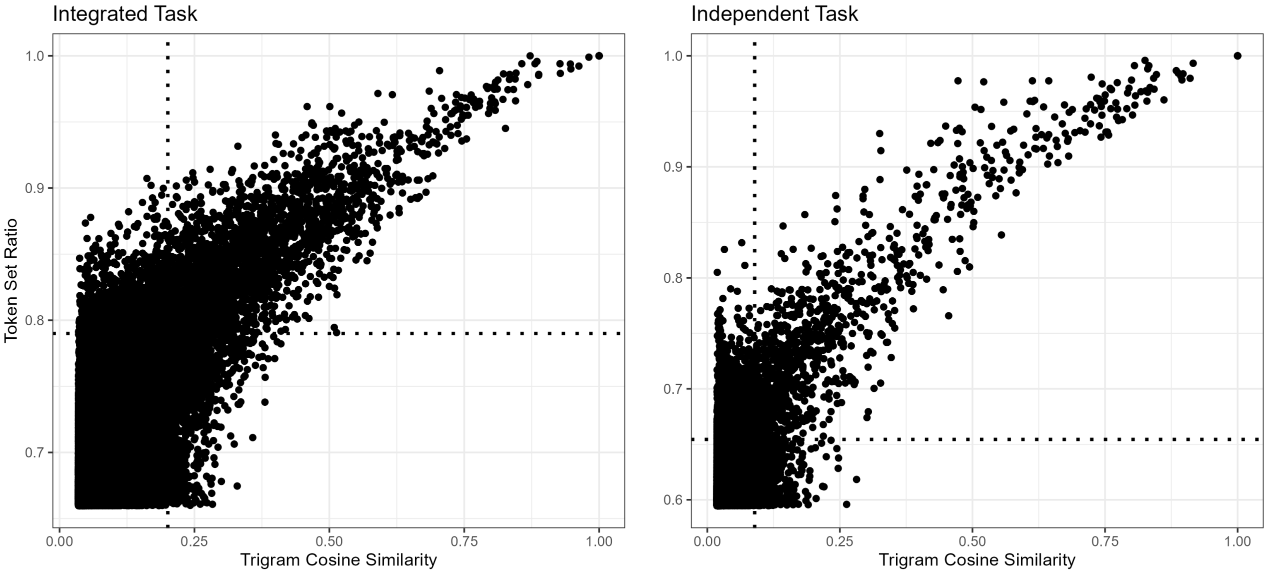


Figure 3: Bivariate distributions of token set ratio and trigram cosine similarity measures from the 99th percentile values by task type. The dotted lines represent the 99.99th percentile values in the respective distributions.

We also examined how long it took to compute the similarity measures for all incoming pairs across administrations. Because administrations differed in terms of the number of total incoming texts (and thus the number of all possible text pairs), this gave us an opportunity to evaluate the computing time requirement at various scales. Figure 4 shows the results in terms of the total computing time (in the left panel) and the per-pair computing time (in the right panel) for integrated task texts on a laptop utilizing a single 1.6 GHz CPU core and 16 GB RAM. Computing time results for independent task texts were highly comparable to those for the integrated task texts and are thus not shown to avoid redundancy. The two panels in Figure 4 present essentially the same information: the computing time increased approximately linearly as a function of the number of pairs. The deviation from a perfectly linear relationship is not surprising because there were other factors that affected the computing time (e.g., the number of unique trigrams in the administration, the number of words in each text within a pair).

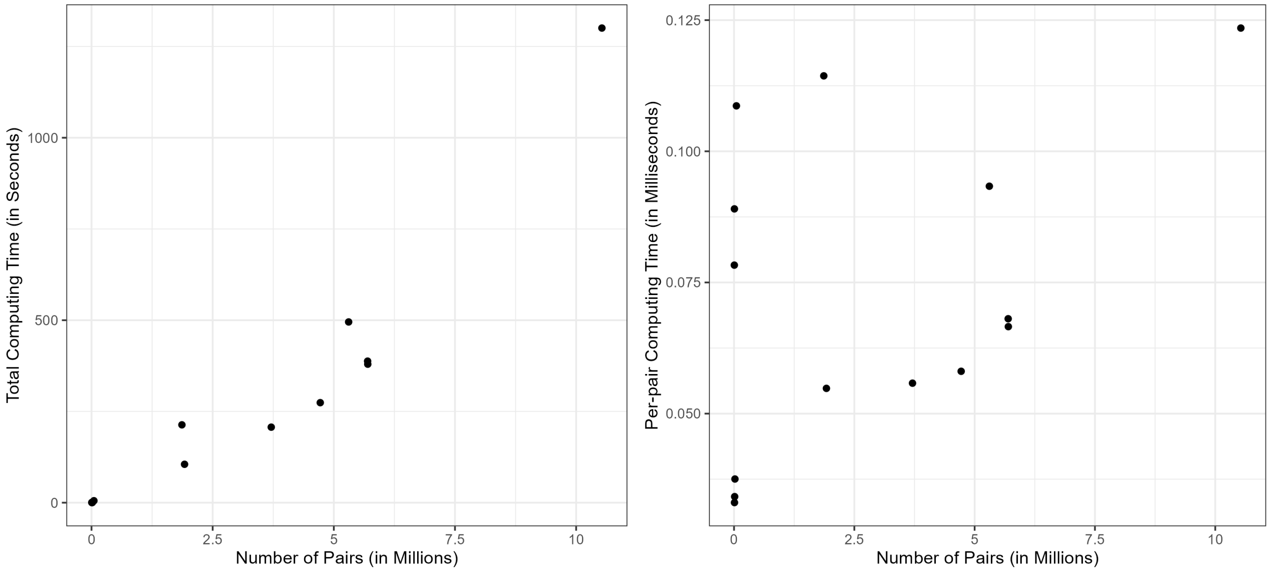


Figure 4: Total computing time and per-pair computing time for all pairwise comparisons among integrated texts. Each dot represents computing time for an administration, whose size (in the number of unique pairs) is shown in the x-axis.

The (approximately) linear relationship provides an empirical justification to obtain a preliminary estimate of computing time for a future administration using linear extrapolation. For example, with the maximum rate of 0.125 millisecond per pair, a large administration with 10000 total incoming texts, which is approximately twice larger than the largest administration (and approximately four times more unique text pairs) would need approximately 6000 seconds to complete all pairwise similarity comparisons on the same laptop. This estimate, which is less than two hours, is a tiny fraction of the operational turnaround time of several days for the TOEFL iBT. Moreover, we expect that, when implemented operationally, the similarity computing process will run on a much more powerful and reliable machine than the laptop we used in this study, which could lead to even shorter per-pair computing time.

5.3.2 Comparing Incoming Texts to Lecture Script

There were a total of 64 lecture scripts in our data set. They varied widely in terms of the number of texts submitted, from 1 to 1209. Because we were mainly interested in extreme values in the similarity measure distributions, we focused on scripts with large number of texts. We thus excluded 44 scripts with fewer than 500 texts and focused on the remaining 20. The empirical distributions of all three similarity measures across the 20 scripts are presented in Figure 5.

The similarity measure distributions were highly comparable across scripts (and within measures) except for the three last ones: Scripts 18 to 20. Specifically, Scripts 18 and 19 yielded more outliers with excessive similarity values than the first seventeen scripts, and Script 20 produced even more excessive similarity outliers as well as an overall distribution that differed from the rest. We examined the text that led to the maximum similarity values for each script and found no evidence of excessive verbatim copying in the texts from the first eighteen scripts. However, we did observe evidence of verbatim copying in the texts from the last three scripts. Moreover, the extent of such evidence was aligned to the magnitude of the similarity measures; the larger the similarity measures, the more extensive verbatim copies were.

The comparison against lecture scripts required much less computing resources than the all-pairwise comparison (shown in Figure 4). This reduction in computing resources had to do with the difference in the number of required comparisons between the two: for the lecture script comparison, we only needed to make as many comparisons as there were incoming texts because each incoming text only needs to be compared to its own lecture script. The computing time did not exceed a minute even for the script with more than a thousand texts. We thus consider the impact of this comparison on the overall score turnaround time trivial.

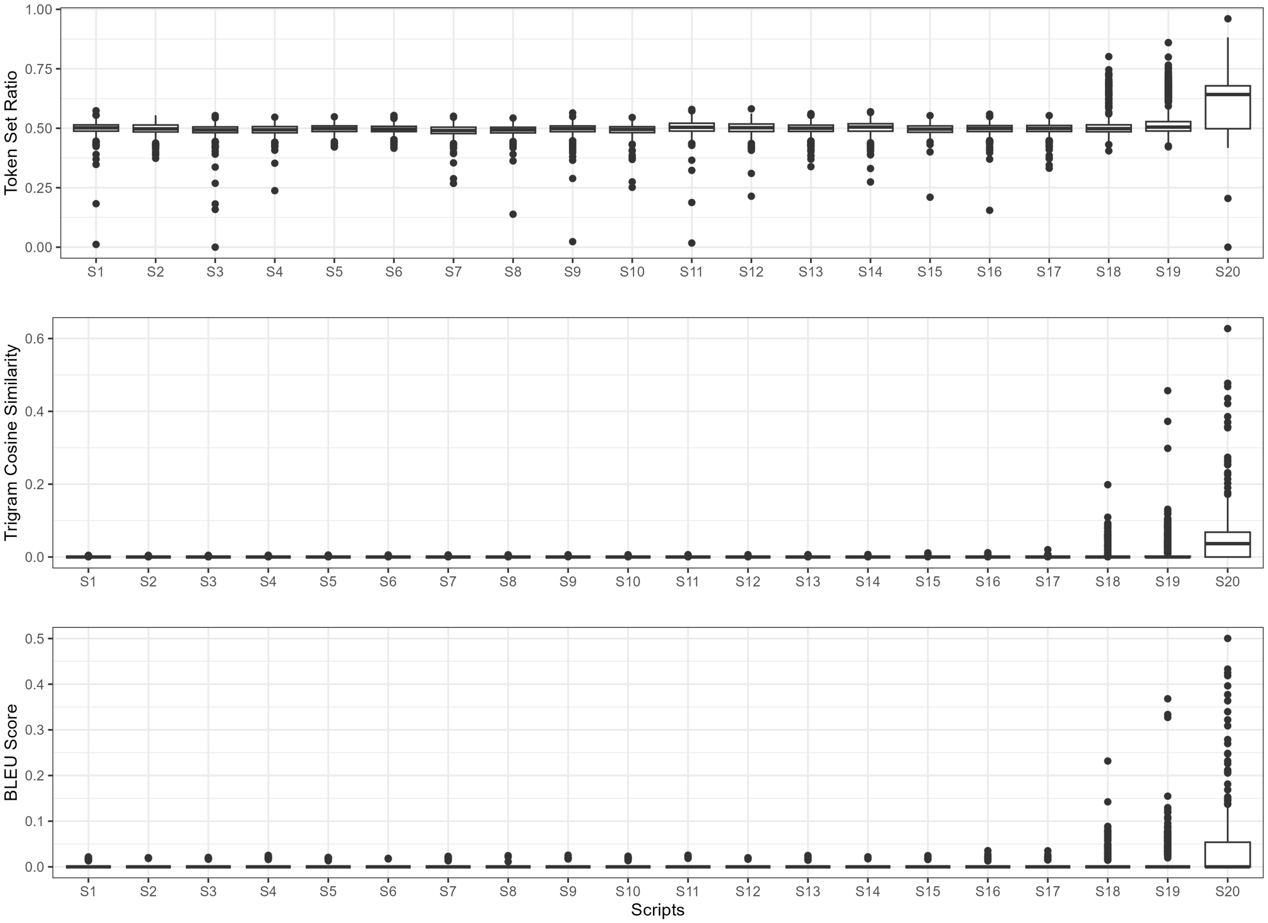


Figure 5: Empirical distributions of token set ratio, trigram cosine similarity, and bleu score measures across the 20 scripts with more than 500 incoming texts. Script numbers are assigned based on the increasing order of the maximum trigram cosine similarity value.

5.3.3 Expert Review of Flagged Text Pairs for Independent Task

We decided to flag independent task text pairs for expert review by utilizing the two measures in a conjunctive manner; that is, a text pair was flagged if and only if it yielded the token set ratio and trigram cosine similarity values larger than their respective threshold values. This choice was motivated by the strong linear relationship between the measures in the relevant right tail region of the bivariate distribution.

We initially picked the threshold values of 0.7 and 0.3 for the token set ratio and trigram cosine similarity measures, respectively. These values corresponded to points beyond which only a very small fraction of text pairs could be found in their respective distributions (as can be seen in Figure 2). Experts in the TOEFL iBT Writing team reviewed the flagged text pairs from the independent tasks and considered all of them as clear cases of non-authentic texts. That is, the AutoESD similarity measures with the initial threshold values achieved perfect precision in detecting suspect independent task text pairs. We then progressively lowered the threshold values for the two measures and provided additional samples for review. The experts noted that, as we lowered the threshold values, the number of false positives also increased, which indicated a close relationship between the AutoESD similarity measures and human expert judgments.

6 Discussions

The findings of the evaluation study provided empirical evidence to gauge how effectively the proposed AutoESD system would detect non-authentic texts for its intended use case. The system was capable of identifying suspect texts based on the distributional characteristics of its similarity measures. The subsequent expert reviews showed that the identified text pairs were indeed similar to each other. Moreover, the detection performance was aligned to the threshold values of the AutoESD similarity measures, indicating that the measures were predictive of human expert judgment. The estimated computing time for operational administrations was short enough not to affect the current score turnaround time at all. We thus consider the overall performance of the AutoESD system as highly encouraging.

The relationship between the detection performance and threshold values can be leveraged to adapt the system to potential changes in available resources. Specifically, the similarity threshold values could be manipulated to find a desirable precision-coverage tradeoff given a fixed amount of available expert review time. For example, if experts could spend more time reviewing suspect texts, the current threshold values can be lowered to achieve even better coverage. On the other hand, if desired, we could raise the threshold values to avoid overwhelming the review process while focusing on highly suspect submissions characterized by extremely large similarity values.

The system can also be refined to improve coverage without sacrificing the precision, or vice versa, by means of a separate classifier (or an ensemble of classifiers) using human expert review results as labels. At the time of the evaluation study, we did not have enough data with expert labels to pursue such a classifier. An important difference between the initial detection of suspect texts and the later, more refined classification based on a separate classifier is that the number of texts that need to be considered by the classifier is expected to be much smaller than that encountered during the initial detection stage. This difference can be leveraged in the design of a classifier by utilizing a wide variety of features, including computationally intensive ones. We believe that a well-performing classifier can provide an impactful added-value to AutoESD by improving the coverage without necessarily increasing the number of reviews needed.

This project was guided by a singular focus on providing a practical yet effective means of identifying suspect texts for large-scale writing tests. The practical focus of this project led to decisions that were made quickly based on the availability of data as well as our experience. For example, the data set we used to evaluate the system performance was limited in that it contained operational responses collected in a short time window. Moreover, the selection of the similarity measures was largely motivated by our prior experience. We acknowledge that those decisions may not have been optimal and consider them as limitations of this project. Another limitation of AutoESD has to do with its dependence upon multiple uses of the same source text; that is, AutoESD is not designed to detect a non-authentic response that is unique within a given administration.

The encouraging results and the limitations of this project provide multiple promising avenues for future work. Although we focused on evaluating the performance of AutoESD on the TOEFL iBT test data set, the similarity measures we used are generically applicable to any pairwise text comparison. The proposed system can thus be effective in detecting non-authentic texts for other standardized writing tests. Another line of future work involves, as noted earlier, refining the system output by training a classifier for even more highly suspect texts. As of this writing, this effort has already begun and yielded promising primary results, and its details will be documented in a separate report. We also regard evaluating other similarity measures and developing new ones as a next step towards making AutoESD more effective and generalizable. Lastly, we consider keystroke logs as a promising source of data to detect unique non-authentic texts. There is empirical evidence for differences in keystroke logs from genuine drafting and those from source copying (Conijin et al., 2019; Deane et al., 2018) and for the potential to exploit the differences to detect non-authentic texts (Trezise et al., 2019) and other types of security breach (Choi et al., 2021).

References

Alzahrani, S. M., Salim, N., & Abraham, A. (2011). Understanding plagiarism linguistic patterns, textual features, and detection methods. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *42*(2), 133–149.

Attali, Y., & Burstein, J. (2006). Automated essay scoring with e-rater® v. 2. *The Journal of Technology, Learning and Assessment*, *4*(3).

Bachmann, M. (2021, October). *Maxbachmann/rapidfuzz: Release 1.8.0* (Version v1.8.0). Zenodo. https://doi.org/10.5281/zenodo.5584996

Britt, M. A., Wiemer-Hastings, P., Larson, A. A., & Perfetti, C. A. (2004). Using intelligent feedback to improve sourcing and integration in students' essays. *International Journal of Artificial Intelligence in Education*, *14*(3-4), 359–374.

Choi, I., Hao, J., Deane, P., & Zhang, M. (2021). *Benchmark keystroke biometrics accuracy from high-stakes writing tasks* [ETS Research Report (ETS-RR-21-15)]. Educational Testing Service.

Conijin, R., Roeser, J., & van Zaanen, M. (2019). Understanding the keystroke log: The effect of writing tasks on keystroke features. *Reading and Writing*, *32*.

Damerau, F. J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM*, *7*(3), 171–176.

Deane, P., Roth, A., Litz, A., Goswami, V., Steck, F., Lewis, M., & Richter, T. (2018). *Behavioral differences between retyping, drafting, and editing: A writing process analysis* [ETS Research Memoradum (ETS-RM-18-06)]. Educational Testing Service.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, *41*(6), 391–407.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *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)*, 4171–4186.

Educational Testing Service. (2023, July). Toefl ibt information bulletin.

El Moatez Billah Nagoudi, A. K., Cherroun, H., & Schwab, D. (2018). 2l-apd: A two-level plagiarism detection system for arabic documents. *Cybernetics and Information Technologies*, *18*(1), 124–138.

Foltỳnek, T., Meuschke, N., & Gipp, B. (2019). Academic plagiarism detection: A systematic literature review. *ACM Computing Surveys (CSUR)*, *52*(6), 1–42.

Gipp, B., Meuschke, N., & Breitinger, C. (2014). Citation-based plagiarism detection: Practicability on a large-scale scientific corpus. *Journal of the Association for Information Science and Technology*, *65*(8), 1527–1540.

Glavaš, G., Franco-Salvador, M., Ponzetto, S. P., & Rosso, P. (2018). A resource-light method for cross-lingual semantic textual similarity. *Knowledge-based systems*, *143*, 1–9.

Hadjieleftheriou, M., & Srivastava, D. (2010). Weighted set-based string similarity. *IEEE Data Eng. Bull.*, *33*(1), 25–36.

Heintze, N. (1996). Scalable document fingerprinting. *1996 USENIX workshop on electronic commerce*, *3*(1).

Hussain, S. F., & Suryani, A. (2015). On retrieving intelligently plagiarized documents using semantic similarity. *Engineering Applications of Artificial Intelligence*, *45*, 246–258.

Levenshtein, V. I. (1965). Binary codes capable of correcting, deletion, insertions, and reversals. *Cybernetics and Control Theory*, *10*, 707–710.

Lyon, C., Malcolm, J., & Dickerson, B. (2001). Detecting short passages of similar text in large document collections. *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing.*

Manber, U. (1994). Finding similar files in a large file system. *Usenix Winter*, *94*, 1–10.

Mozgovoy, M., Kakkonen, T., & Sutinen, E. (2007). Using natural language parsers in plagiarism detection. *Workshop on Speech and Language Technology in Education.*

Nahnsen, T., Uzuner, O., & Katz, B. (2005). *Lexical chains and sliding locality windows in content-based text similarity detection* (tech. rep.). AIM-2005-017. CSAIL Memo.

Needleman, S. B., & Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, *48*(3), 443–453.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 311–318.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2227–2237.

Salton, G., & Buckley, C. (1988). Weighting approaches in automatic text retrieval. *Information Processing and Management*, *24*(5), 513–523.

Shivakumar, N., & Garcia-Molina, H. (1995). Scam: A copy detection mechanism for digital document. *Proceedings of the Second International Conference on the Theory and Practice of Digital Libraries.*

Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, *28*(1), 11–21.

Tetreault, J., & Chodorow, M. (2010). *Essay similarity detection.* Unpublished internal report.

Trezise, K., Tracii, R., de Barba, P., & Kennedy, G. (2019). Detecting academic misconduct using learning analytics. *Journal of Learning Analytics*, *6*(3).

Uzuner, O., Katz, B., & Nahnsen, T. (2005). Using syntactic information to identify plagiarism. *Proceedings of the Second Workshop on Building Educational Applications Using NLP*, 37–44.

Wang, X., Evanini, K., Qian, Y., & Zechner, K. (2019). Using very deep convolutional neural networks to automatically detect plagiarized spoken responses. *2019 IEEE Automatic Speech Recognition and Understanding Workshop*, 764–771.