Content Generation by Variational Encoders

The purpose of this collection of content generation software is to create
synthesized but plausbly 'genuine' examples of multiple text and media genre.
There are several criteria used in the choice of methods or combination of
methods, and various categories of text or media require differing approaches.
In the initial research the three categories covered are published scholarly
documents and abstracts, emails collected by the W3C on topic discussion
threads, scam 'phishing' emails, and Twitter 128-character 'tweets'.

The basic methodology is to start with a collection, or 'corpus', of real
examples and then modify these examples to produce a novel synthesized set.
The first goal is to pass routine machine scrutiny by not appearing in any
database of collected real-world examples. This is achieved by ensuring that
no member of the synthesized set matches any previous 'recorded' example from
the world. This goal is achieved for emails and tweets by breaking the texts
as well as possible into sentences, or separable phrases, cleaning the
sub-texts of possible meaningless clutter and symbols, and identifying key
nouns and adjectives in the sub-text and substituting a synonym drawn
automatically from the Python Natural Language Toolkit (nltk) module massive
set of synonym/antonym collection 'Wordnet'. Finally, in some cases, some
hand editing is applied to correct 'poor synonym choices' made by applying
the Wordnet ranked sets automatically.

The three categories chosen span a broad spectrum of language usage.
Tweets are the least grammatical or even 'rational' language uses, and consist
of exclamations, abbreviated and letter expressions, and all sorts of near
nonsense. However, the synonym substitution works best on tweets, and with
(almost) no need of hand-editing because the grammatical

expectations for
tweets are near zero so almost any textual abberartion can be dismissed as
being a 'typo' or 'tweet-slang' or just subjective craziness.

Emails are similar to tweets in that often there are suspensions of normal word
choice and grammar, but generally the overall intention of emails, due to their
sometimes much greater length than tweets, is to convey a unified package
and flow of ideas and intentions. For this reason much more hand-editing of
synonyms is needed to produce plausible instances. Further, since emails
function at a higher and more lengthy level of continuous discourse more
sphisticated means can be applied after simple synonym substitution, or
neglecting synonym substitution altogether. Thus emails can be regared in most
cases as cogent 'documents', and the same methods used to synthesize published
documents (discussed next) can be used on emails also.

In the case of published documents and abstracts preserving correct spelling
and correct grammar is essential. In almost all case, neural networks, whether
based on character or word based probability models, make egregious
errors in either spelling (in the character case) or grammar (in the word
case). For this reason the methodology selected is an original approach
though borrowing techniques from document classification schemes and
dataset principle components analysis (without dimension reduction).
Documents are broken into sentences and then processed to omit semantically
inert 'stop-words' such as 'the', 'and', etc. Further all remaining words in
the sentences are 'stemmed', i.e elimination tense suffixes, plurals, etc.,
leaving just the meaningful core of each term. Then a huge matrix is created
in which the rows are each sentence in every document in the corpus, and the
columns are each filtered stem term in any document in the corpus. This is
called a 'document-term' matrix and is huge, typically hundreds of sentences

and thousands of terms. The entries in each cell (i,j) are the number of occurrences of the j'th term in the i'th sentence, divided by the frequency of occurrence of the j'th term in all documents. The resulting numbers in the cell represent usage of the term within the sentence but weighs much more heavily unique or characteristic terms, and de-emphasizes more generally used terms. The key step is the encoder part of the Variational Encoder.
A Singular Value Decomposition (https://towardsdatascience.com/understanding-singular-value-decomposition-and-its-application-in-data-science-388a54be95d)
of the document-term matrix produces a product of three matrices, the left matrix representing each sentence as the weighted sum of feature vectors, and providing a semantic distance metric among all sentences. For each document in the corpus, for each (or many) sentences in the docuemnt, the sentence is replaced by another sentence, not in the document but in the corpus, which closely matches the sentence in meaning. By this method all documents maintain correct spelling and grammar, and may only be detectable as being synthesized by a slight nuance of semantic flow - probably undetectable by software, and probably overlooked even by human scrutiny.

Here is a diagram of the two systems, in which tweets and emails are processed by the first (vaeet) by synonym substitution, and emails and published documents are processed by the second (vaed). Note: 'vae' stands for 'variational auto-encoder, 'et' for emails-tweets, and 'd' for documents.

The semantic system (vaed - vae_proto) relies on creating a 'document-term' matrix in which the sentences of all documents are represented by the number of occurences of each word in the union of all words in the corpus.
(actually it is more than that - the frequency of the term in the

sentence is
divided by the frequency in the corpus - this weighs common words less)
(even more so-called stop-words ('the', 'a' etc.) are eliminated and a few
other more natural language processing techniques even more 'nit-picky')

However the main theoretic principle is that the huge matrix of sentences by
terms can be decomposed into a product of three matrices - the first associating
each sentence with a vector in a big space of features, which provides a way
to measure the semantic distance of one sentence with another.

Choosing a similar semantically related sentence for one or many in the document
creates a new document passing machine scrutiny by novelty, and passing human
scrutiny by correct spelling and grammar - with perhaps some notion of strange
semantic discontinuity - but not often or egregious - so probably passing human
scrutiny also.