ONOMÁZEIN
Journal of linguistics, philology and translation

PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
FACULTAD DE LETRAS

A methodology for synthetic corpus engineering

Carlos Periñán-Pascual
Universitat Politècnica de València
España

ONOMÁZEIN 70 (December 2025): 242-262
DOI: 10.7764/onomazein.70.11
ISSN: 0718-5758

Carlos Periñán-Pascual: Applied Linguistics Department, Escuela Politécnica Superior de Gandia, Universitat Politècnica de València, España. ORCID: 0000-0002-6483-4712. | E-mail: jopepas3@upv.es

Received: November, 2024
Accepted: July, 2025

Abstract

This article describes a methodological framework for developing synthetic corpora with a small language model from a prompt engineering perspective. Instead of the typical approach in text mining, which prioritises corpus size in model training, this study applies a corpus linguistics methodology that accounts for corpus domain and distribution considerations to generate diverse and realistic texts. These synthetic corpora were evaluated through their integration into a text classification system to detect social problems. Therefore, the objective is to demonstrate whether using a theoretically sound methodology based on corpus linguistics can improve the performance of systems trained with such synthetic corpora. The study concludes that factors such as stratification and proportionality in the sampling method have even more impact than corpus size.

Keywords: synthetic corpus engineering; corpus linguistics; language model; text classification; social problem.

1. Introduction

Recent advances in large language models (LLMs) have driven rapid progress in natural language generation systems capable of producing high-quality text in various tasks, including translation, summarisation, creative writing, and question answering. Incorporating task-specific constraints directly into the prompt, i.e. a set of natural language instructions and examples given to the LLM to guide the output, can customise the content and style of generated text. One application of controllable text generation with LLMs is synthetic corpus generation. It should be noted that corpora can be constructed for two different purposes. On the one hand, we can develop corpora for linguistic research. In this case, as explained by Egbert and others (2022), the corpus is intended to help researchers achieve generalisable empirical descriptions of language use in a real-world discourse domain; in other words, corpora permit generalisations about a particular domain (i.e. the population, in statistical terms) from the analysis of linguistic features based on a collection of texts (i.e. the sample). Since corpora should provide sufficient evidence for researchers' purposes, corpus size is usually considered the key factor for determining corpus representativeness concerning the needs of the research project (Corpas Pastor and Seghiri, 2010). However, Egbert and others (2022) stated that increasing corpus size does not always result in better resources, mainly when there is selection bias (i.e. a skewing that favours the inclusion of some particular types of texts over others) or coverage bias (i.e. a mismatch between the texts that are found in the real-world domain and the texts included in the corpus). On the other hand, we can develop corpora for training, validating and evaluating machine-learning models for text mining. In this case, unlike linguistic corpora, most researchers do not design such resources with solid methodological underpinnings but simply apply the well-known approach *the more, the better*. However, some researchers (Gatta and others, 2014; Lang and others, 2016; Yang and others, 2022) have suggested that this approach should be revised and improved with alternative text selection techniques.

In this article, we devised a methodology for developing synthetic training corpora, adapted from Egbert and others (2022) framework of linguistic corpus design. To this end, we employed prompt engineering techniques with a small language model (SLM) rather than an LLM to facilitate its integration into downstream applications[1]. Our synthetic corpora were evaluated based on their performance within a text classifier for community-problem detection. In particular, we tested the impact of corpus size and other considerations (e.g. sampling method) on the classifier. The remainder of this article is organised as follows.

1 Based on Minaee and others (2024), we use SLM to refer to small (less than 1 billion parameters) and medium-sized (between 1 and 10 billion parameters) language models and LLM to refer to large (between 10 and 100 billion parameters) and very large (i.e. more than 100 billion parameters) language models.

Section 2 describes the most relevant works for this study. Section 3 presents the case study of our experiment, and section 4 focuses on the proposed methodology. Finally, we evaluate the results of the experiment in section 5 and present some conclusions in section 6.

2. Constructing synthetic corpora

Constructing synthetic corpora should be explored as a particular task in constrained text generation (Garbacea and Mei, 2022), also known as controllable text generation (Zhang and others, 2023), which includes dialogue generation, text summarisation, text simplification, and storytelling, among other tasks. Constrained text generation focuses on "generating coherent and logical texts that do (not) cover lexical concepts [...] desired to be (not) present in the output, as well as generate outputs that abide to specific format, semantic, syntactic or utility rules to reflect the particular interests of the system user" (Garbacea and Mei, 2022: 2). In short, the system should be able to "generate texts that meet certain controllable constraints that are imposed by the targeted applications and users" (Zhang and others, 2023: 2). Indeed, according to Garbacea and Mei (2022), only imposing constraints on the output can successfully achieve valuable text generation for real-world applications. In other words, models can produce realistic, coherent, and fluent texts if we control the attributes of the generated output. Humans can easily impose restrictions on naturally generated sentences by using their commonsense reasoning ability. However, composing realistically plausible sentences in the presence of constraints remains a challenge for machine-learning models.

Although generating synthetic texts with language models is not new (cf. Anaby-Tavor and others, 2020; Puri and others, 2020), it has recently gained great interest due to the latest advances in transformer-based language models pre-trained on vast quantities of text, which have significantly contributed to the emergence of LLMs such as GPT-4 (OpenAI, 2023), LLaMA (Touvron and others, 2023), and PaLM (Chowdhery and others, 2023), among many others. Indeed, the superior generative capacity of LLMs has led to a revolutionary breakthrough in deep learning and natural language processing (NLP). As Zhang and others (2023) noted, methods based on such models are becoming the mainstream research in constrained text generation. In the last few years, LLMs have significantly enhanced their text generation capabilities and extended their applicability to a broader range of NLP tasks. To comprehend and produce natural language expressions, LLMs employ complex neural networks containing billions of parameters trained on large corpora from the web using a self-supervised learning approach to discover language patterns and subtleties (Raiaan and others, 2024). Once LLMs are trained, their main advantage is that they can be reused to perform tasks for which they had not been explicitly trained, thus contributing to resource adaptability. However, the process by which LLMs learn to produce texts of different semantic and pragmatic features is not transparent. LLMs are considered black

boxes, as their internal decision-making processes are often not easily interpretable, even to the experts who designed them. As Berber Sardinha (2024) explained, the ability of LLMs to produce human-like texts emerges from underlying mathematical algorithms and model weights during the training phase instead of being explicitly instructed to mimic specific linguistic features.

LLMs can impose constraints on the desired output through prompting (or prompt-based learning), which has become a way of fine-tuning LLMs. Prompts can contain instructions, questions, and some optional elements (e.g. context, examples, etc.), which can guide the model to achieve the optimal response. Therefore, prompting can be understood as a form of programming (Reynolds and McDonell, 2021) since it allows researchers to instruct computers in natural language. This method introduces the need for *prompt engineering*, whose goal is to find the most appropriate prompt for a given LLM to solve the task (Liu and others, 2023). For example, LLMs can perform zero-shot learning, where they are directly prompted to perform the task, and few-shot learning, where a few real-world text instances are provided as examples. However, due to the black-box characteristics of LLMs, the controllability of such models becomes a challenging field of research, which is why applying prompt-based learning to transformer-based LLMs for constrained text generation has become a hot academic topic (Zhang and others, 2023).

In this context, OpenAI has significantly contributed to developing LLMs with remarkable proficiency in generating high-quality texts. For example, ChatGPT is based on GPT-3.5 and GPT-4, where the former has been trained with 175 billion parameters and the latter with 1.8 trillion parameters. It should be noted that the ability of LLMs to comprehend intricate language patterns and produce appropriate human-like texts is critically affected by the number of parameters with which the models were trained. The more parameters LLMs have, the better they can learn when prompted. However, ChatGPT poses two problematic issues for researchers. On the one hand, ChatGPT is constantly being updated, so different versions of the model can yield different results. In turn, this issue raises two serious problems over time: (a) researchers do not have control over the performance of the model, and (b) they cannot replicate the results of their experiments. On the other hand, OpenAI does not provide free API services for GPT-3.5 and GPT-4. Therefore, the cost of using any of these models in a production environment can be prohibitive if vast amounts of data have to be processed. For these reasons, Minaee and others (2024) explained that, although LLMs can get better accuracy and performance, there is a current research trend to develop SLMs as a cost-effective alternative in tasks that do not require the capability of LLMs.

3. Case study

The research in this study is conducted in the ALLEGRO project (**A**daptive mu**L**ti-domain socia**L**-media s**E**nsin**G** f**R**amew**O**rk), a general-purpose multi-modal system for the devel-

opment of real-time crowdsensing applications that can accurately reconstruct the state of society as interpreted by the collective intelligence of social media users. In other words, we intend to analyse user-generated content to construct models that can detect community problems in cities, thus considering online users as witnesses of a given society. One of the key modules in ALLEGRO is DIAPASON (unifie**D** hybr**I**d **A**p**P**roach to microtext **A**nalysis in **S**ocial-media cr**O**wdse**N**sing) (Periñán-Pascual, 2023), where English and Spanish texts from social media can be analysed by integrating NLP, machine and deep learning, and knowledge engineering techniques. DIAPASON is provided with an ontology that organises the knowledge extracted from user-contributed data based on community problems that can affect some, most or all citizens. For each problem type in the ontology, we manually constructed a formal language-independent problem schema, which contains the shared conceptual knowledge the community has about the problem type. For example, one of these specific types of community problems is InstitutionalElderAbuse, which is defined as the abuse of older people in institutions and whose problem schema is as follows:

> (elderly-107943870 & (yell-200912833 | threaten-200871405 | humiliate-201799794 | isolate-200495808 | blow-101173038 | shove-100113726 | neglect-100739270) & (doctor-110020890 | nurse-110366966 | social_worker-110620027 | nursing_home-103528100 | hospital-103540595)).

As can be seen, the main elements in problem schemas are concepts and logical operators presented through expressions, whose scope is determined by round brackets. Concepts take the form of WordNet synsets (Fellbaum, 1998), i.e. sets of synonymous words[2]. It should be noted that, although English words in problem schemas are expendable since lexical units can be automatically retrieved from synsets, they are included to facilitate readability. In the problem schema of InstitutionalElderAbuse, the concepts were selected to focus on the main semantic components of the description of this problem type, as shown in table 1.

4. Methodology

4.1. Establishing the research question

The first step in our methodological framework for developing synthetic corpora with pre-trained language models is to establish the research question, which will guide the construction of the corpus. In this case, we want to create several synthetic corpora for the problem type InstitutionalElderAbuse so that each corpus can help DIAPASON automat-

2 WordNet is a lexical database that includes nouns, verbs, adjectives, and adverbs organised into synsets, where each synset represents a distinct concept connected to other synsets through lexical and semantic relations.

TABLE 1

Components of the problem schema of INSTITUTIONALELDERABUSE

ROLE		CONCEPTS
To whom?		elderly-107943870
What?	Psychological abuse	yell-200912833, threaten-200871405, humiliate-201799794, isolate-200495808
	Physical abuse	blow-101173038, shove-100113726
	Neglect	neglect-100739270
Who?		doctor-110020890, nurse-110366966, social_worker-110620027
Where?		nursing_home-103528100, hospital-103540595

ically recognise Spanish user-generated content in the evaluation dataset that could be considered an instance of this community problem.

4.2. Addressing the domain considerations of the corpus

As any corpus is designed to represent a particular domain of language use, one of the critical issues is corpus representativeness, which is determined by two major types of considerations:

a) whether the corpus represents the entire range of texts and text types in the domain of interest (i.e. domain considerations) and

b) whether the corpus represents the distribution of the linguistic features of interest (i.e. distribution considerations).

Whereas domain considerations are described in the remainder of this section, distribution considerations are explained in section 4.3.

Domain considerations relate to those qualitative characteristics of the domain that determine which texts should be included in the corpus, dealing with issues such as domain description and sampling design. On the one hand, the domain description includes identifying the methods and resources for describing the problem type and defining its boundaries (i.e. the type of texts that fall within the domain). In the DIAPASON ontology, problem types result from exploring a variety of knowledge sources, from reports, encyclopedias and textbooks, where problem types are described from a global perspective, to research articles and other scholarly publications, where particular aspects of problems are examined in depth. From such knowledge sources, we organised an inventory of problem types and then created the problem schema of each problem type. The domain of the corpora for the

experiment in this study was the abuse of older people in institutions, where we focused on user-generated content posted on social media[3].

On the other hand, the sampling design requires specifying the sampling unit and the sampling method, which are used to select the specific texts to include in the corpus. In our study, the sampling unit was the text of a post. Regarding the sampling method, which plays an essential role in the textual diversity of the corpus, Li and others (2023) concluded that increasing data diversity is one potential way to improve the effectiveness of synthetic datasets. In this regard, the sampling method involves making decisions on three issues that contribute to increasing textual diversity: stratification, proportionality, and randomness. First is stratification, or the process of generating texts based on specific categories (or strata); that is, we generated texts from each stratum rather than generating texts directly from the entire domain. In particular, we stratified the corpus according to the following criteria:

- Language. In this study, the texts were generated in *Spanish*.
- Name of the problem type. In our experiment, the subject matter of the texts was the *abuse of older people in institutions*.
- Description of the problem type. We described the problem type through lexical contexts, which take the form of sets of keywords. The construction of lexical contexts from a single problem schema is described in section 4.4.
- Figure of speech. Figurative language is a frequent phenomenon in human communication, occurring in spoken and written discourse. Figurative language generation is a text-to-text generation task intended to reformulate a given text into a different one containing the desired figure of speech (e.g. *simile*, *metaphor*, *irony*, and *sarcasm*)[4], while still being faithful to the original context (Lai and Nissim, 2024). Figurative language generation based on LLMs through prompt engineering has been lately explored in various studies (e.g. Bhavya and others, 2022; Chakrabarty and others, 2022; Mittal and others, 2022).
- Function of speech. As we intend to generate posts as claims on community problems, we focused on two functions of speech: *complaint* and *criticism*, which result in statements with different linguistic features (Murphy and Neu, 1996). In complaints, "the speaker expresses displeasure or annoyance as a reaction to past or ongoing action, the consequences of which affect the speaker unfavorably" (Olshtain and Weinbach,

3 See Jiménez-Briones and Felices Lago (2024) for a description of the problem type InstitutionalElderAbuse.

4 As explained by Joshi and others (2016: 6, 129), "irony is a situation in which something which was intended to have a particular result has the opposite or a very different result", whereas "sarcasm is a form of verbal irony that is intended to express contempt or ridicule".

1987: 195). Trosborg (1994: 311-312) defined the complaint as "an illocutionary act in which the speaker (the complainer) expresses his/her disapproval, negative feelings etc. towards the state of affairs described in the proposition (the complainable) and for which he/she holds the hearer (the complainee) responsible, either directly or indirectly". As can be seen, both definitions emphasise that speakers reflect their feelings in complaints. In contrast, criticism draws attention to what is wrong rather than the speaker, making responses much more blunt, contemptuous, and direct (Sauer, 2000).

- Style. According to Kirszner and Mandell (2003), the style used when speaking or writing can be *formal*, *informal*, *colloquial*, and *slang*.
- Tone. This study focused on politeness (i.e. *polite*, *politeness-neutral*, and *rude*), a fundamental aspect of socio-communicative interaction. Indeed, several studies have explored the relationship between politeness and criticism in producing utterances (Lerman, 2006; Henry and Ho, 2010; Masjedi and Paramasivam, 2018).
- Length of texts. We aimed to produce moderate-length posts (i.e. *between 25 and 36 words*) in our experiment. To avoid truncating the output, we decided that the maximum number of tokens to generate should be 120, considering that one word corresponds to approximately two tokens in Spanish.

To summarise, table 2 organises the strata employed in constructing synthetic corpora and presents the values for each stratum in our experiment.

Second, proportionality refers to the distribution of texts within each stratum. In our study, there was an equal number of texts for each value in a given stratum.

Third, randomness is aimed at guaranteeing unbiased samples, as a random sample of texts from the target domain is likely to have less selection bias than a non-random sample. On the one hand, randomness was primarily controlled through the hyperparameter tuning of the SLM. In particular, tweaking the settings of temperature and sampling technique (e.g. Top-K and Top-P) helps get different results for each prompt. In this regard, the higher the values of such settings, the more diverse and creative the output (e.g. in our study, Temperature: 0.9; Top-K: 50; Top-P: 0.9). On the other hand, how the prompt was formulated also contributed to randomness. For example, three representative keywords (i.e. the lexical context) were used to describe the problem type InstitutionalElderAbuse in the prompt. However, we did not impose these keywords on the output (e.g., by inserting the *following keywords must be included in the post* in the prompt) as this strategy could have generated lexically biased texts.

4.3. Addressing the distribution considerations of the corpus

Distribution considerations relate to the quantitative characteristics of the corpus. As we did not intend to construct corpora for linguistic research, we did not orient the study to

TABLE 2

Stratification of synthetic corpora

TYPE	STRATUM	CODE	VALUE
language	language	G	Spanish
semantics	problem-type name	N	abuse of older people in institutions
	problem-type description	D	*[lexical context]*
pragmatics	speech function	U0	criticism
		U1	complaint
	style	S0	formal
		S1	informal
		S2	colloquial
		S3	slang
	tone	T0	polite
		T1	politeness-neutral
		T2	rude
	speech figure	I0	-
		I1	simile or metaphor
		I2	irony or sarcasm
format	text length	L	between 25 and 36 words

particular linguistic features on which frequency analyses could have been conducted. Therefore, we only addressed the issue of corpus size, where details are presented in the following section.

4.4. Constructing the corpus

We experimented with Microsoft Orca 2 7b (Mitra and others, 2023)[5], an open-source instruction-based SLM constructed from the foundation model LLaMA 2 and trained with 7 billion parameters. Microsoft Orca 2 was developed to enhance the ability of SLMs to reason, whose goal was not only to teach smaller models how to use various reasoning techniques but also

5 https://huggingface.co/microsoft/Orca-2-7b.

to help such models decide when to use the most effective reasoning strategy for the task at hand. As a result, Microsoft Orca 2 is suitable as a reasoning engine over the knowledge provided to the model through in-context instructions (i.e. prompting). Indeed, Mitra and others (2023) demonstrated that Microsoft Orca 2 outperforms models of similar size and performs similarly to models that are five to ten times larger, especially on zero-shot tasks.

Regarding prompting, we constructed a template whose blanks were programmatically filled in with the values from the different strata so that various customised prompts could be generated to guide the SLM. We adopted the zero-shot paradigm, where the SLM is fine-tuned only with a direct specification of the task to make the model more readily adaptable to new problem types. The prompt template was as follows:

> *You are a citizen who publishes posts on social networks about community problems in your city. Write one post [text_length] in [language] as a [speech_function] about the community problem "[problem_type]", which is described with the following keywords: "[lexical_context]". Write the post in [style] style and [tone] tone. [speech_figure] Your response must only include the post. Remember, your post should fall within the word count range specified above. Post:*

The placeholders in this template are enclosed in square brackets and filled in with the values in table 2[6]. It should be recalled that lexical contexts in the form of keywords play a critical role in shaping the output of prompts. The procedure used to automatically construct the lexical contexts that describe a given problem type in the DIAPASON ontology is as follows:

a) We obtain a list of all the synsets included in the problem schema (i.e. *source synsets*). The problem schema of INSTITUTIONALELDERABUSE has thirteen source synsets, as shown in table 1.

b) We create the *conceptual profile* of each source synset. A conceptual profile takes the form of a list of paradigmatically related synsets (i.e. *expanded synsets*). This step consists of three tasks:

   b1) Each source synset is expanded through different WordNet relations according to the following pipelines (*relation-1* > *relation-2* > .... *relation-n*), where each step in the pipeline applies the expansion on all the synsets derived from previous steps (i.e. source or expanded): *hyponym* > *pertain_to* > *related_to* for nouns, *hyponym* > *subevent* > *related_to* for verbs, *near_synonym* > *pertain_to* > *is_derived_from* > *related_to* for adjectives, and *is_derived_from* for adverbs.

6 It should be noted that, when *speech figure* was instantiated, the actual value was *Describe the problem using [a simile or metaphor | irony or sarcasm].*

b2) We obtain synset paraphrases for each source synset and its expanded synsets, only allowing synset paraphrases with the same category as the synsets from which they were derived (that is, the words linked to the source synset and those linked to the expanded synsets should have the same part of speech). Synset paraphrases, also considered as expanded synsets, are retrieved from our Synset-based Paraphrase Database, which was automatically constructed by leveraging resources such as the Paraphrase Database (Ganitkevitch and others, 2013), LessLex (Colla and others, 2020), and WordNet.

b3) Based on LessLex embeddings, we discard any WordNet- and paraphrase-based expanded synset whose cosine similarity with the synset from which it was derived is below a particular threshold. In our study, the threshold was 0.9. Therefore, the conceptual profile of each source synset is an array of expanded synsets ranked according to their cosine similarity with the source synset.

c) We map each conceptual profile into a *lexical profile*. To this end, each synset in conceptual profiles is lexicalised into one or more lexical units through WordNet, only allowing lexical units with a frequency higher than two in the reference corpus (i.e. BNC for English[7] and CORPES for Spanish[8]). To illustrate, we present the conceptual profile and the Spanish lexical profile of the source concept *blow-101173038*:

101173038 = { 101173038, 100134780, 101176031, 100134391, 100133338 }
101173038 = { golpe, golpetazo, impacto, bofetada, gancho, paliza, pinchazo, punch, puñetazo, puñete, tortazo, golpazo, trastazo, zurra, batazo, castañazo, cate, leñazo, porrazo }

d) We create the *conceptual contexts* of the problem schema. This step consists of two tasks:

d1) We derive all statements of source synsets from the problem schema (i.e. *source conceptual contexts*). For example, here are some of the thirty-five source conceptual contexts from the problem schema of InstitutionalElderAbuse:

[elderly-107943870, shove-100113726, doctor-110020890]
[elderly-107943870, shove-100113726, hospital-103540595]
[elderly-107943870, neglect-100739270, nurse-110366966]
[elderly-107943870, neglect-100739270, nursing_home-103528100]
etc.

7 https://www.kilgarriff.co.uk/bnc-readme.html.
8 https://apps2.rae.es/corpes/.

d2) From each source conceptual context, we generate *expanded conceptual contexts* through a data-augmentation process. The following two techniques are sequentially applied: synset swap (i.e. two synsets swap their positions in the source conceptual context) and synset replacement (i.e. a synset in the source conceptual context is replaced with one paradigmatically related synset from its conceptual profile). In our study, 1,046 conceptual contexts were constructed: 35 source conceptual contexts + 1,011 expanded conceptual contexts (i.e. 105 augmented with synset swap and 906 with synset replacement).

e) We map each conceptual context into multiple *lexical contexts* by replacing each synset in the conceptual context with one lexical unit from the lexical profile of the synset. In our study, we obtained 5,787 Spanish lexical contexts.

In the case of InstitutionalElderAbuse, if we had constructed a synthetic corpus using all the Spanish lexical contexts, we would have produced 8,333,280 texts just for a single problem type. As this would have generated an unwieldy corpus, we focused on a tiny fraction of the entire corpus by experimenting with only ten lexical contexts from the conceptual contexts. Table 3 shows these lexical contexts, which finally generated a corpus size of 720 texts.

TABLE 3

Spanish lexical contexts used in the prompt template

LEXICAL CONTEXT
tercera edad, dejadez, hospital
tercera edad, negligencia, doctor
tercera edad, facultativo, desamparo
viejo, residencia de mayor, humillar
tercera edad, residencia de mayor, gritar
viejo, empujón, médico
separar, tercera edad, asistente social
mayor, bofetada, clínico
anciano, aislar, clínica
enfermero, desidia, viejo

5. Evaluation

5.1. Experiment

We evaluated the quality of synthetic corpora by measuring the performance (i.e. Precision, Recall and F1 score) of a text-mining system for community-problem detection. To this end, we compiled a test corpus of Spanish tweets, which resulted from queries based on a combination of the following keywords related to INSTITUTIONALELDERABUSE: *abuso*, *anciano*, *maltrato*, *residencia*, and *tercera edad*. The test corpus comprised 160 annotated tweets (i.e. 78 positive and 82 negative).

We adopted an unsupervised text classification approach, using synthetic corpora to train self-organising maps (SOMs) (Kohonen, 2001). Typically, a SOM is a two-dimensional single-layer neural network used to reduce the dimensionality of high-dimensional datasets while preserving the topological structure of data. In particular, our approach to perform community-problem detection based on SOMs is as follows:

a) We train a separate SOM for each problem type so that each SOM can learn the patterns specific to the corpus of that problem type. In this article, we only focused on INSTITUTIONALELDERABUSE.

b) When a post from the test corpus needs to be classified, we pass its text embedding through each of the trained SOMs and calculate the similarity score of the post to the Best Matching Unit (BMU) in each SOM. To this end, we use the Gaussian kernel as the similarity measure, which is based on the squared Euclidean distance. In particular, this measure can convert Euclidean distances [0, ∞] into similarity scores [0, 1], so it can be thought of as a non-linear normalisation function of the Euclidean distance, being computed as follows:

$$K(X,Y) = \exp\left(-\frac{||X-Y||^2}{2\sigma^2}\right) = \exp\left(-\frac{D(X,Y)}{2\sigma^2}\right)$$

where the value of sigma is 1.

c) The new post is assigned the problem types whose similarity scores are above a threshold.

5.2. Results

We classified the Spanish tweets in the test corpus with SOMs trained with eight synthetic corpora. Table 4 shows the information about each corpus.

The values used in the *Categories* and *Distribution* columns in table 4 can be found in the *Code* column in table 2. The parameters used in the training of the SOMs are shown in table 5.

TABLE 4

Distribution of texts in the synthetic corpora

TYPE	TEXTS	CATEGORIES	DISTRIBUTION
1	720	USTI	360 U0 + 360 U1, where each U has 90 S0 + 90 S1 + 90 S2 + 90 S3, each S has 30 T0 + 30 T1 + 30 T2, and each T has 10 I0 + 10 I1 + 10 I2
2	720	U	360 U0 + 360 U1
3	720	UI	360 U0 + 360 U1, where each U has 120 I0 + 120 I1 + 120 I2
4	720	US	360 U0 + 360 U1, where each U has 90 S0 + 90 S1 + 90 S2 + 90 S3
5	720	UT	360 U0 + 360 U1, where each U has 120 T0 + 120 T1 + 120 T2
6	1,440	UI	Corpus 3 combined with Corpus 2
7	2,160	USI	Corpus 6 combined with Corpus 4
8	2,880	USTI	Corpus 7 combined with Corpus 5

TABLE 5

Parameters in the training of SOMs

PARAMETER	VALUE
Width x Height	12x12 (720 texts), 14x14 (1,440 texts), 16x16 (2,160 texts), 18x18 (2,880 texts)
Learning rate	0.5
Learning radius	6 (720 texts), 7 (1,440 texts), 8 (2,160 texts), 9 (2,880 texts)
Iterations	8,000

We used the following procedure for each synthetic corpus and its corresponding SOM. First, we computed the similarity score of each post in the test corpus to the BMU in the SOM. Second, we considered each of the 160 similarity scores (i.e. one for each tweet in the test corpus) as a potential cut-off, where values above the cut-off were considered positive and those below the cut-off as negative. Third, we computed Precision, Recall and F1 for each cut-off value and finally selected the optimal threshold based on the value that provided the best F1 score. Table 6 presents the values corresponding to the optimal threshold in classifying the tweets with Microsoft Orca 2 7b, where Type represents a given synthetic corpus in table 4.

TABLE 6

Text classification with Microsoft Orca 2 7b

TYPE	THRESHOLD	PRECISION	RECALL	F1
1	0.662	0.770	0.859	0.812
2	0.647	0.829	0.808	0.818
3	0.628	0.828	0.923	0.873
4	0.649	0.802	0.987	0.885
5	0.652	0.807	0.910	0.855
6	0.628	0.826	0.910	0.866
7	0.650	0.816	0.910	0.861
8	0.649	0.771	0.949	0.851

5.3. Discussion

We can draw several conclusions from analysing the data in table 6. On the one hand, when we compare the performance of the classifier with corpora stratified in different degrees while preserving their size (Types 1-5), we conclude that the corpus that only focused on the speech function (Type 2) has worse performance than those corpora where the speech function goes with another constraint, for example, the style, the tone, and the figurative language (Types 3-5). Therefore, stratification has a positive impact. However, the speech function only and, mainly, the speech function with another constraint provide better results than all the constraints together. This should be interpreted as meaning that stratification is relevant, but not all constraints are equally important. On the other hand, when we compare the performance of the classifier with corpora of different sizes and stratification (Types 6-8), we realise that increasing corpus size does not guarantee better performance, particularly when proportionality is disregarded. In fact, it can be seen that some smaller corpora with a few constraints (Types 3 and 4) provide better results than corpora that are two, three or four times larger. Therefore, the sampling method employed to construct synthetic corpora is a key determinant of the performance of classifiers, where factors such as stratification and proportionality have more impact than corpus size. Moreover, considering the range of optimal threshold values in Table 6, we note that the difference between minimum and maximum values is very small. This fact is also positive, as choosing the average value of the corresponding range (i.e. 0.65) can guarantee that the margin of error with respect to the optimal threshold for each synthetic corpus is ±0.02.

6. Conclusions

Pre-trained transformer-based language models are currently becoming the mainstream in controllable text generation. One of the applications of this research field is what we call *synthetic corpus engineering*, where such language models are instructed to automatically create high-quality text to be primarily used in the training of text-mining systems. In this context, we devised a methodological framework for developing synthetic corpora, where an SLM was chosen as a cost-effective way to integrate a generative model into downstream applications. To test the quality of synthetic corpora, we implemented a text classifier for community-problem detection. In particular, we employed synthetic corpora to train SOMs that could identify whether a given user-generated content is related to the problem type InstitutionalElderAbuse (i.e. the abuse of older people in hospitals and nursing homes). Our experiment demonstrated that factors such as stratification and proportionality in the sampling design of synthetic corpora have a more significant impact on the performance of classifiers than corpus size.

It should be noted that this unsupervised text-classification approach is efficient for community-problem detection, as this task involves multi-label text classification with hundreds of classes based on a dynamic ontological model, which is likely to be updated with a growing number of new problem types. In this case, it is evident that retraining the entire machine-learning model from scratch or using techniques such as incremental learning (Wolters and others, 2022) is inadequate for dealing with a large inventory of classes. Instead, we can perform a cascade of binary classification tasks based on the ontological hierarchy using multiple independent SOMs as classifiers, so we need one machine-learning model for each problem type in the ontology and one distinct corpus to train each model. However, developing multiple fine-grained training corpora of natural texts in such a dynamic environment is not feasible. Therefore, automatically constructing synthetic corpora can help solve this issue, providing that they are developed with a sound methodology.

Finally, while this study offers valuable contributions, several limitations must be addressed in future research. These include: (a) testing the proposed methodological framework with different languages and problem types; (b) increasing the size of synthetic corpora by experimenting with a wider range of lexical contexts, which play a key role in shaping the output of prompts; and (c) evaluating the quality of synthetic corpora using a larger test dataset of manually annotated tweets. In addition, a significant challenge was identified. In particular, there is a need to develop a more efficient procedure for automatically determining the optimal similarity-based threshold for a given synthetic corpus, as the current approach is limited to selecting the cut-off value that provides the highest F1 score.

7. Acknowledgements

This publication is part of the R&D&I projects PID2020-112827GB-I00 and PID2023-147137NB-I00, funded by MICIU/AEI/10.13039/501100011033 and by ERDF, EU, and partially supported by the research project CIPROM/2023/29, funded by "Direcció General de Ciència i Investigació" Generalitat Valenciana (Spain).

8. References

Anaby-Tavor, Ateret, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, Naama Tepper and Naama Zwerdling, 2020: "Do not have enough data? Deep learning to the rescue!", *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, 7383-7390, doi:10.1609/aaai.v34i05.6233.

Berber Sardinha, Tony, 2024: "AI-generated vs human-authored texts: A multidimensional comparison", *Applied Corpus Linguistics* 4 (1), 1-10, doi:10.1016/j.acorp.2023.100083.

Bhavya, Bhavya, Jinjun Xiong and ChengXiang Zhai, 2022: "Analogy generation by prompting large language models: A case study of InstructGPT", *Proceedings of the 15th International Conference on Natural Language Generation*, 298-312, doi:10.18653/v1/2022.inlg-main.25.

Chakrabarty, Tuhin, Yejin Choi and Vered Shwartz, 2022: "It's not rocket science: Interpreting figurative language in narratives", *Transactions of the Association for Computational Linguistics* 10, 589-606, doi:10.1162/tacl_a_00478.

Chowdhery, Aakanksha, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton and Sebastian Gehrmann and others, 2023: "PaLM: Scaling language modeling with pathways", *Journal of Machine Learning Research* 24, 1-113.

Colla, Davide, Enrico Mensa and Daniele P. Radicioni, 2020: "LessLex: Linking multilingual embeddings to sense representations of LEXical items", *Computational Linguistics* 46 (2), 289-333, doi:10.1162/coli_a_00375.

Corpas Pastor, Gloria, and Miriam Seghiri, 2010: "Size matters: A quantitative approach to corpus representativeness" in Rosa Rabadán, Trinidad Guzmán and Marisa Fernández (eds.): *Lengua, traducción, recepción*. En honor de Julio César Santoyo, León: Universidad de León, 111-145.

Egbert, Jesse, Douglas Biber and Bethany Gray, 2022: *Designing and evaluating language corpora: A practical framework for corpus representativeness*, Cambridge: Cambridge University Press, doi:10.1017/9781316584880.

Fellbaum, Christiane (ed.), 1998: *WordNet: An electronic lexical database*, Cambridge: MIT Press, doi:10.7551/mitpress/7287.001.0001.

Ganitkevitch, Juri, Benjamin Van Durme and Chris Callison-Burch, 2013: "PPDB: The Paraphrase Database", *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 758-764.

Garbacea, Cristina, and Qiaozhu Mei, 2022: "Why is constrained neural language generation particularly challenging?", *arXiv preprint*, doi:10.48550/arXiv.2206.05395.

Gatta, Roberto, Mauro Vallati, Berardino de Bari and Mahmut Ozsahin, 2014: "The impact of different training sets on medical documents classification", *Proceedings of the 3rd International Conference on Artificial Intelligence and Assistive Medicine*, 1-5.

Henry, Alex, and Debbie G. E. Ho, 2010: "The act of complaining in Brunei—Then and now", *Journal of Pragmatics* 42 (3), 840-855, doi:10.1016/j.pragma.2009.08.011.

Jiménez-Briones, Rocío, and Ángel Felices Lago, 2024: "La formalización del conocimiento en DIAPASON a través de una muestra de problemas poblacionales y macroeconómicos" in Françoise Olmo-Cazevieille (ed.): I*nvestigación lingüística en entornos digitales*, Valencia: Tirant Lo Blanch, 187-217.

Joshi, Aditya, Vaibhav Tripathi, Pushpak Bhattacharyya, Mark James Carman, Meghna Singh, Jaya Saraswati and Rajita Shukla, 2016: "How challenging is sarcasm versus irony classification? A study with a dataset from English literature", *Proceedings of the Australasian Language Technology Association Workshop* 2016, 123-127.

Kohonen, Teuvo, 2001: *Self-Organizing Maps*, 3rd ed., Berlin-Heidelberg: Springer, doi:10.1007/978-3-642-56927-2].

Lai, Huiyuan, and Malvina Nissim, 2024: "A survey on automatic generation of figurative language: From rule-based systems to large language models", *ACM Computing Surveys*, doi:10.1145/3654795.

Lang, Frederique, Diego Chavarro and Yuxian Liu, 2016: "Can automatic classification help to increase accuracy in data collection?", *Information Science* 1 (3), 42-58, doi:10.20309/jdis.201619.

Lerman, Dawn, 2006: "Consumer politeness and complaining behavior", *Journal of Services Marketing* 20 (2), 92-100, doi:10.1108/08876040610657020.

Li, Zhuoyan, Hangxiao Zhu, Zhuoran Lu and Ming Yin, 2023: "Synthetic data generation with large language models for text classification: Potential and limitations", *Proceedings of*

the 2023 Conference on Empirical Methods in Natural Language Processing, 10443-10461, doi:10.18653/v1/2023.emnlp-main.647.

Liu, Pengfei, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi and Graham Neubig, 2023: "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing", *ACM Computing Surveys* 55 (9), 195: 1-35, doi:10.1145/3560815].

Kirszner, Laurie G., and Stephen R. Mandell, 2003: *Analysis of language style*, New York: Winthrop.

Masjedi, Narges, and Shamala Paramasivam, 2018: "Complaint and politeness strategies used by Iranian speakers of English", *International Journal of Applied Linguistics & English Literature* 7 (4), 38-49, doi:10.7575/aiac.ijalel.v.7n.3p.38.

Minaee, Shervin, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao, 2024: "Large language models: A survey", *arXiv preprint*, doi:10.48550/arXiv.2402.06196.

Mitra, Arindam, Luciano del Corro, Shweti Mahajan, Andres Codas, Clarisse Simoes, Sahaj Agarwal, Xuxi Chen, Anastasia Razdaibiedina, Erik Jones, Kriti Aggarwal, Hamid Palangi, Guoqing Zheng, Corby Rosset, Hamed Khanpour and Ahmed Awadallah, 2023: "Orca 2: Teaching small language models how to reason", *arXiv preprint*, doi:10.48550/arXiv.2311.11045.

Mittal, Anirudh, Yufei Tian and Nanyun Peng, 2022: "AmbiPun: Generating humorous puns with ambiguous context", *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1053-1062, doi:10.18653/v1/2022.naacl-main.77.

Murphy, Beth, and Joyce Neu, 1996: "My grade's too low: The speech act set of complaining" in Susan M. Gass and Joyce Neu (eds.): *Speech acts across cultures: Challenges to communication in second language*, Berlin: Mouton de Gruyter, 191-216, doi:10.1515/9783110219289.2.191.

Olshtain, Elite, and Liora Weinbach, 1987: "Complaints: A study of speech act behavior among native and non-native speakers of Hebrew" in Jef Verschueren and Marcella Bertuccelli Papi (eds.): *The Pragmatic Perspective*, Amsterdam: John Benjamins, 195-208, doi:10.1075/pbcs.5.15ols.

OpenAI, 2023: "GPT-4 technical report", *arXiv preprint*, doi:10.48550/arXiv.2303.08774.

Periñán-Pascual, Carlos, 2023: "From Smart City to Smart Society: A quality-of-life ontological model for problem detection from user-generated content", *Applied Ontology* 18 (3), 263-306, doi:10.3233/AO-230281.

Puri, Raul, Ryan Spring, Mohammad Shoeybi, Mostofa Patwary and Bryan Catanzaro, 2020: "Training question answering models from synthetic data", *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 5811-5826, doi:10.18653/v1/2020.emnlp-main.468.

Raiaan, Mohaimenul Azam Kha, Md. Saddam Hossain Mukta, Kaniz Fatema, Nur Mohammad Fahad, Sadman Sakib, Most Marufatul Jannat Mim, Jubaer Ahmad, Mohammed Eunus Ali and Sami Azam, 2024: "A review on large language models: Architectures, applications, taxonomies, open issues and challenges", *IEEE Access* 12, 26839-26874, doi:10.1109/ACCESS.2024.3365742.

Reynolds, Laria, and Kyle McDonell, 2021: "Prompt programming for large language models: Beyond the few-shot paradigm", *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, 1-7, doi:10.1145/3411763.3451760.

Sauer, Melanie, 2000: "Complaints: A cross-cultural study of pragmatic strategies and linguistic forms", *The 2000 American Association For Applied Linguistics Conference*.

Touvron, Hugo, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave and Guillaume Lample, 2023: "LLaMA: Open and efficient foundation language models", *arXiv preprint*, doi:10.48550/arXiv.2302.13971.

Trosborg, Anna, 1994: *Interlanguage pragmatics: Requests, complaints, and apologies*, Berlin: Walter de Gruyter, doi:10.1515/9783110885286.

Wolters, Anna, Kilian Müller, and Dennis M. Riehle, 2022: "Incremental machine learning for text classification in comment moderation systems", *Proceedings of the 4th Multidisciplinary International Symposium on Disinformation in Open Online Media*, 138-153, doi:10.1007/978-3-031-18253-2_10.

Yang, Heming, Ke Yang and Erhan Zhang, 2022: "Brand celebrity matching model based on natural language processing", *arXiv preprint*, doi:10.48550/arXiv.2208.08887.

Zhang, Hanqing, Haolin Song, Shaoyu Li, Ming Zhou, and Dawei SONG, 2023: "A survey of controllable text generation using transformer-based pre-trained language models", *ACM Computing Surveys* 56 (3), 1-37, doi:10.1145/3617680.