Unsupervised Grouping of Public Procurement Similar Items: Which text representation should I use?

Pedro P. V. Brum, Mariana O. Silva, Gabriel P. Oliveira,
Lucas G. L. Costa, Anisio Lacerda, Gisele L. Pappa
Computer Science Department, Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
{pedrobrum, mariana.santos, gabrielpoliveira}@dcc.ufmg.br
lucas-lage@ufmg.br, {anisio, glpappa}@dcc.ufmg.br

Abstract

In public procurement, establishing reference prices is essential to guide competitors in setting product prices. Group-purchased products, which are not standardized by default, are necessary to estimate reference prices. Text clustering techniques can be used to group similar items based on their descriptions, enabling the definition of reference prices for specific products or services. However, selecting an appropriate representation for text is challenging. This paper introduces a framework for text cleaning, extraction, and representation. We test eight distinct sentence representations tailored for public procurement item descriptions. Among these representations, we propose an approach that captures the most important components of item descriptions. Through extensive evaluation of a dataset comprising over 2 million items, our findings show that using sophisticated supervised methods to derive vectors for unsupervised tasks offers little advantages over leveraging unsupervised methods. Our results also highlight that domain-specific contextual knowledge is crucial for representation improvement.

Keywords: sentence representation, text clustering, enhanced embeddings

1. Introduction

Public procurement is vital in governmental operations, ensuring fair competition, transparency, and fiscal responsibility in acquiring goods and services from private entities. This process encourages competition among companies and ensures that governments receive value for money while upholding ethical standards and combating corruption.

Ideally, there should be a standardized system for naming the items—goods or services—procured, which would facilitate comparing reference prices across different procurement instances. Such standardization is crucial for detecting overpricing and uncovering irregularities. However, it falls short of this ideal scenario. In 2022 alone, the Brazilian federal government disbursed over 80 billion dollars in public procurement,[1] with comparable expenditures at the state and municipal levels.

Given the lack of uniformity in naming items and the absence of a common information system spanning all levels of government procurement, identifying whether two item descriptions are the same becomes crucial. Although such a task can be tackled as a problem of entity disambiguation (Godény, 2012), it presents unique challenges due to the unstructured, highly specific, and informally written nature of item description texts. While most methods are designed to handle traditional grammatical structures, text data with idiosyncratic characteristics pose distinct challenges (Reimers and Gurevych, 2019; Yong and Torrent, 2020).

One of the key challenges is recognizing item attributes, which often include colour, material, model, and numerical quantities, such as the number of items required. For instance, the description *"plastic mask with elastic band and layer of polypropylene box 100 units"* highlights the complexity of the task, as it contains specific characteristics of the purchased item. Finding an appropriate representation for items is paramount to the problem of item description disambiguation.

Using embeddings to represent sentences has been widely employed in recent works (Reimers and Gurevych, 2019). Most of these methods calculate the average or sum of the learned vector for each word to represent sentences. However, this approach has a significant drawback: it assigns the same weight to all words in a sentence. In grammatically correct sentences, the subject, predicate, and object are known to be the most critical parts as they convey the meaning of the primary sentence.

In item or product descriptions, nouns are more important for defining the items and measurement units, while numbers are essential for specifying sizes and quantities. For instance, distinguishing between boxes containing 1,000 or 100 masks requires attention to numbers. To address this issue, methods for word representation have been modified to learn sentence representation (Reimers and Gurevych, 2019), where weights for words in a sentence are learned automatically. Other approaches aim to enhance sentence representation models, such as Sentence-BERT (Reimers and Gurevych, 2019), by incorporating post-tagging or named en-

[1] `https://bit.ly/procurementbudget`

LREC-COLING 2024, pages 17176–17185
20-25 May, 2024.

tity recognition (Yin et al., 2020).

Once we find an appropriate representation, we can address the problem of item description disambiguation in different ways. Here, we use a clustering approach to group similar items based on their descriptions. Note that learning representations for unsupervised tasks is not as common as for supervised tasks. Therefore, this work compares a set of representation models, including an enhanced domain-dependent embedded text representation, intended for use in unsupervised contexts to distinguish unique items.

We test eight different representations, ranging from a simple Bag-of-Words to Sentence-BERT, using a dataset comprising over 2.1 million item descriptions written in Portuguese. Among these representations, we adapt the component-focused representation proposed by Yin et al. (2020) to item representation. This approach divides the original description/sentence into the complete text and the component-focused segment of the sentence. We then separately learn embeddings for the component-focused structures and subsequently integrate them with the embeddings of the complete text. While Yin et al. (2020) based their component-focused structure on dependency parsing, we adapted it to our problem, where the component-focused structure includes nouns, units of measure, and numerical values.

To evaluate these representations, we introduce a generic framework called AFFAIR (**A** **F**ramework **F**or generAting **I**tem **R**epesentaions). Our framework contains three steps, including text cleaning and information extraction, and it also provides a systematic approach to generate item representations for analysis and clustering.

The main contributions of this paper are summarized as follows:

1. The adaptation of the component-focused method proposed in Yin et al. (2020) to deal with items/products;

2. A comparison of eight different sentence model representations for items that do not follow a formal grammar structure;

3. A framework to deal with the pre-processing and model representation of structures of texts that do not follow the formal grammar structure and deal with numbers.

2. Related Work

Text representation is an area of research broadly studied in natural language processing (NLP) and is currently one that most benefits from using deep neural networks (Devlin et al., 2019; Melamud et al., 2016). For a long time, the most commonly used strategy for building embedded representations for sentences/documents was to average word embeddings, such as word2vec (Mikolov et al., 2013) or Global Vectors (GloVe) (Pennington et al., 2014b).

Recently, researchers have explored alternative strategies to derive document embeddings, often utilizing pre-trained models in an unsupervised manner (Conneau et al., 2017b; Reimers and Gurevych, 2019; Yong and Torrent, 2020). These models, known as encoders, pose key questions: which neural network architecture is optimal for the target task, and how should the network be trained? Many unsupervised methods like SkipThought (Kiros et al., 2015), FastSent (Hill et al., 2016), and BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) have been employed for document encoding, though supervised learning approaches have garnered attention more recently.

One notable supervised method is Sentence-BERT (SBERT), which modifies BERT using a Siamese network structure to produce semantically meaningful sentence embeddings (Reimers and Gurevych, 2019; Wang et al., 2022). SBERT can be fine-tuned on context-specific data and used to map sentences to a vector space in an unsupervised manner, which clustering algorithms and other machine-learning models can use.

Furthermore, large language models, such as SBERT, have been enhanced in different ways for specific tasks. For example, Chen et al. (2021) proposed Inductive Document Representation Learning (IDRL) to enhance the representations of short texts. It maps short text structures into a graph network and recursively aggregates neighbor information of the words in the unseen documents. Karami et al. (2022) also proposed a method that uses punctuation to enhance text representations.

Yin et al. (2020) introduced Component Focusing (CF)-BERT, which divides the input sentence into two segments: the basic part S_{basic} and the component-enhanced part S_{cf}. While S_{basic} retains complete sentence information, S_{cf} focuses on critical sentence elements, such as subject, predicate, and object, obtained through dependency parsing.

Compared to the studies mentioned above, the proposed framework and its representation method (E-SBERT) were specifically designed to generate domain-aware representations from non-structured text, while accounting for the unique requirements of clustering tasks. The literature on domain-aware representations is still limited, especially concerning unstructured text, and our approach fills this gap by building upon the model of CF-BERT.

While CF-BERT traditionally defines domain-relevant components based on dependency parsing, we adapted this concept to accommodate unstructured text. Instead of relying on formal gram-

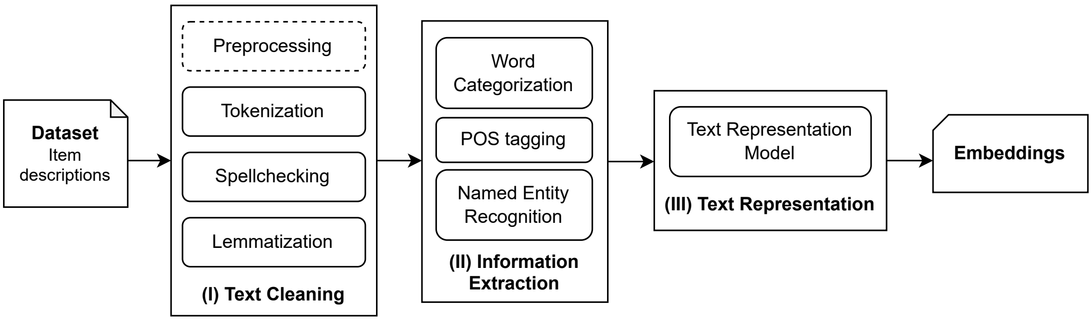

Figure 1: AFFAIR overview.

mar structures, our model focuses on extracting named entities and part-of-speech tags to capture domain-specific information. This novel approach allows us to investigate the effectiveness of leveraging domain knowledge to enhance representation models for unstructured data, predominantly pretrained on structured datasets.

3. AFFAIR: A Framework For generAting Item Repesentations

This section describes AFFAIR, our proposed framework for evaluating different representations for public procurement items. The framework works in three main steps, as shown in Figure 1: (i) Text Cleaning, (ii) Information Extraction, and (iii) Text Representation. The framework's output is embeddings suitable for any target task, such as identifying unique items in an unsupervised manner. Each step comprises a set of operations, with optional steps indicated by dotted boxes.

Following preprocessing, tokenization breaks down preprocessed descriptions into individual tokens, while spellchecking corrects any misspelt words using the Levenshtein distance. Spellchecking is exclusively applied to words not found in a dictionary,[2] where the algorithm identifies similar words with a distance of up to two operations and replaces the original word with the most similar match. Lemmatization replaces each token with its canonical form using a dictionary of inflected Portuguese words,[3] helping standardize words appearing in various forms.

Given the significance of digits in item representations, we also normalize all numbers by representing them in scientific notation, combining an exponent and a potency (e.g., 314.0 is represented as 3.14e+02). This approach was motivated by findings presented in (Zhang et al., 2020).

In the context of text representation enhancement, we also consider the method of canonizing numbers proposed by (Zhang et al., 2020). The authors showed that BERT's numerical reasoning ability is limited when dealing with small-magnitude numbers. Thus, they proposed replacing every number in the data with its representation in scientific notation, introducing a new token ([EXP]) to denote the exponent. Such a strategy enables BERT to associate sentence objects with their corresponding magnitudes expressed in the exponent, leading to notable improvements in results.

3.1. Information Extraction

The proposed framework extracts relevant information from the output of the set of tokens by the text cleaning phase. Two approaches are used for this: POS tagging and Named Entity Recognition (NER). For POS tagging and NER, we used the library spaCy,[4] which provides models pre-trained on the OntoNotes corpus.[5]

The significance attributed to words identified by each tool varies depending on the application. In the context of public procurement items, *nouns* are deemed most critical, as determined through an initial characterization of the real-world dataset, instead of the subject, verb, and object as would typically be the case in formal text. For NER, we focus on eight out of nineteen named entity categories: PERSON, NOR (nationalities or religious or political group), PRODUCT, LANGUAGE, PERCENT, QUANTITY (measurements, such as weight or distance), ORDINAL and CARDINAL (numerals not falling under another type).

Words identified within these eight NER categories are used to construct a structured representation of the text, comprising six categories: units of measurement, colors, materials (e.g., wood, plastic), numbers (i.e., all numeric terms), size (e.g., small, large), and quantity (i.e., terms describing the item's presentation form or quantity, e.g., package, unit). The mapping between NER categories and the structured representation is established

[2] https://bit.ly/CorpusNILC
[3] https://bit.ly/DELAF_PB
[4] https://spacy.io
[5] https://catalog.ldc.upenn.edu/LDC2013T19

Table 1: Overview of the evaluated sentence embedding representations.

Model	Architecture	Input Level	Embedding size
Bag-of-Words	-	Words	vocabulary size
GloVe (Average) (Pennington et al., 2014a)	-	Words	300
fastText (Joulin et al., 2017)	MLP	Words	300
SIF (Weighted Average) (Arora et al., 2017)	MLP	Words	300
Sent2Vec (Pagliardini et al., 2018)	MLP	Sentences	700
InferSent (Conneau et al., 2017a)	BiLSTM	Sentences	1024
Sentence-BERT (Reimers and Gurevych, 2019)	BERT	Sentences	728
Enhanced-SBERT (Proposed)	**CF-BERT**	**Sentences**	**728**

manually. Any term not belonging to the above categories is categorized as "text".

To illustrate, consider the preprocessed item description *"adhesive tape autoclave 19 mm x 30 m"*. After the Information Extraction step, the following word categories would be identified: *description: {adhesive, tape, autoclave, x}*; *units of measure*: *{mm, m}*; and *numbers*: *{19, 30}*. Regarding POS tags, the category labels would be assigned as follows: *Noun*:*{tape, autoclave}* and *Adjective: {adhesive}*. As for NER, the category labels would be assigned as follows: *PRODUCT*:*{tape, adhesive}* and *CARDINAL:{19, 30}*.

3.2. Text Representation

The final step involves building vector representations for the item descriptions, leveraging the structured descriptions provided by the information extraction module as input for a model to learn embeddings. Here, the framework explores eight distinct sentence representations. Table 1 presents an overview of the compared methods, detailing the architecture type for DNN-based models, the input data type (words or sentences), and the dimensions of the embeddings they were trained with.

The last entry of the table introduces Enhanced-SBERT (E-SBERT), a method we adapted from the principles of CF-BERT (Yin et al., 2020), which integrates SBERT to enrich sentence vectors by incorporating domain-dependent components. E-SBERT is crucial in focusing on essential item attributes facilitated by the information extraction module. By structuring information based on relevant NER categories (e.g., units of measurement, colors, materials), E-SBERT identifies and prioritizes domain-dependent knowledge, including nouns and words associated with these categories.

When generating the component-focused embedding, all nouns and words identified as belonging to relevant NER categories are concatenated and used to enhance the representation. Currently, these words receive equal weight in the embedding generation process. However, future versions of the framework could explore assigning different weights to specific categories based on their importance in describing item attributes. For instance, attributes like color or size might be more critical than quantity in certain contexts.

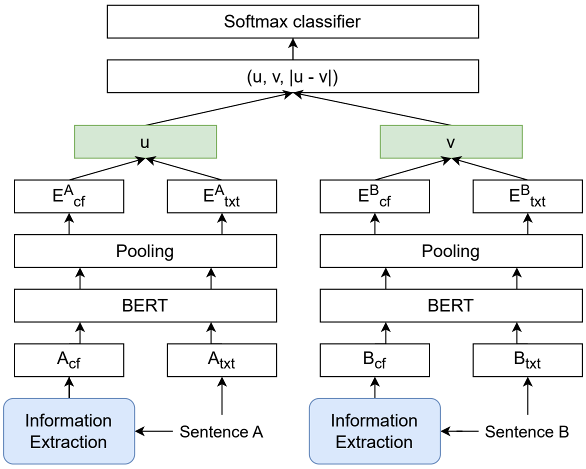

Figure 2: Enhancement architecture for learning enhanced embedding in a classification task.

Figure 2 illustrates the structure of the enhanced representation model. Two input sentences, A and B, are initially passed to the model to generate fixed-size document embeddings considering the whole text. In parallel, the model performs information extraction for each sentence to obtain their component-enhanced segments and generate their embeddings. Hence, each item description yields two vectors: the embeddings for the complete text of the sentence E_{txt} and the embedding for the component-enhanced part E_{cf}. The final embedding e for each sentence is then computed as:

$$e = \sigma \times E_{txt} + E_{cf},$$

where σ is a weight factor that adjusts the ratio of the component-enhanced part embedding to generate the final sentence representation. Such a hyperparameter requires tuning during training and controls the contribution of the component-enhanced part to the final representation. A σ value closer to 0 indicates heavier reliance on the complete text, while a value closer to 1 suggests greater reliance on the component-enhanced part.

In Figure 2, the resulting embeddings are represented by u and v, which can act as input for downstream NLP tasks. For a classification task, as considered here, u and v are concatenated with the element-wise difference $|u - v|$ subsequently processed through a softmax function. The cross-entropy loss is optimized in this case. Note that the important components of a sentence can be easily adapted to different tasks.

4. Identifying Groups of Similar Items

This section outlines the methodology employed to identify groups of descriptions referring to the same item, which is the primary objective of this study. Here, as the datasets we consider have millions of items, we employ a combination of heuristics and more sophisticated clustering methods to handle the computational complexity efficiently.

Generally, the initial tokens of a description contain the most pertinent information about the item. For example, for the item “syringe for insulin injection”, the token “syringe” is the most relevant term in the description. Hence, one straightforward heuristic is to group items sharing the same first token. In practice, all items beginning with “syringe” would be grouped. While this approach applies to Spanish or Portuguese, it can be adapted to suit the grammatical nuances of other languages.

The *first token* heuristic (FT) is a starting point for the clustering algorithm. However, it has drawbacks. First, descriptions with misspellings or closely joined words may inadvertently form initial groups with only one matching item. Second, items with similar descriptions but referring to different objects may be erroneously grouped, as seen in examples like “milk powder” and “milk shaker”, which are both included in the "milk" group. Nonetheless, this behavior is expected, as it motivates the need for more sophisticated clustering approaches.

Following the application of the FT heuristic, the embeddings generated by the representation methods are normalized using the UMAP method (McInnes et al., 2018). Next, the preliminary groups formed by the FT heuristic are refined using HDBSCAN (McInnes et al., 2017). Each sentence is represented in this phase by its respective generated embedding method. For each preliminary group, HDBSCAN is executed, considering the Euclidean distance between sentence embeddings.

Several factors guided the choice of HDBSCAN: (i) it can handle clusters with different densities and detect outliers (i.e., noise); (ii) it can automatically determine the number of clusters, which is particularly useful when dealing with large datasets; (iii) it is a hierarchical algorithm that provides a more realistic representation of the data structure, allowing users to explore different levels of granularity in the resulting clustering.

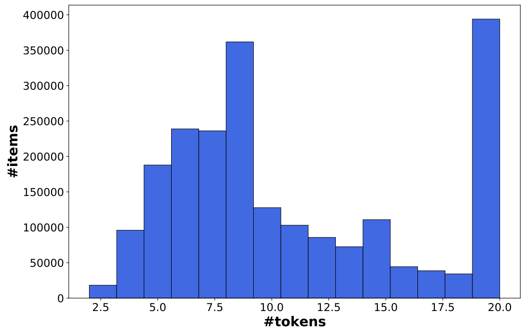

Figure 3: Item descriptions length distribution.

5. Experimental Evaluation

This section compares the model representations integrated into AFFAIR, as outlined in Table 1, for clustering public procurement items. First, we detail the real-world dataset considered (Section 5.1) and the setup of hyperparameters (Section 5.2). Next, we present the evaluation metrics used to assess the clustering results (Section 5.3). Finally, we discuss the experimental results (Section 5.4).

5.1. Dataset

We consider a dataset of real-world items collected from the Prosecution Service of the Brazilian state of Minas Gerais system and written in Portuguese. The dataset contains 196,747 public procurements held between 2015 and 2018, spanning various public administration sectors and encompassing diverse types of items. Here, we focus on items with descriptions containing at least one numeric term or unit of measure, as these components are central to our study. Such a selection resulted in 2,149,533 items, of which 2,096,664 are unique exact descriptions. Figure 3 shows the distribution of the number of tokens per item description.

5.2. Hyperparameters Setup

The size of the embeddings for each model was defined based on preliminary studies, with default parameter values used when not explicitly defined. For the Bag-of-Words model, embedding size varies according to the dataset, specifically the vocabulary size. For pre-trained word embedding methods such as GloVe, fastText, and SIF, the vector size is set to 300.

Regarding other hyperparameters, in fastText, the context window size is set to 10, with negative sampling employing five negative examples and an

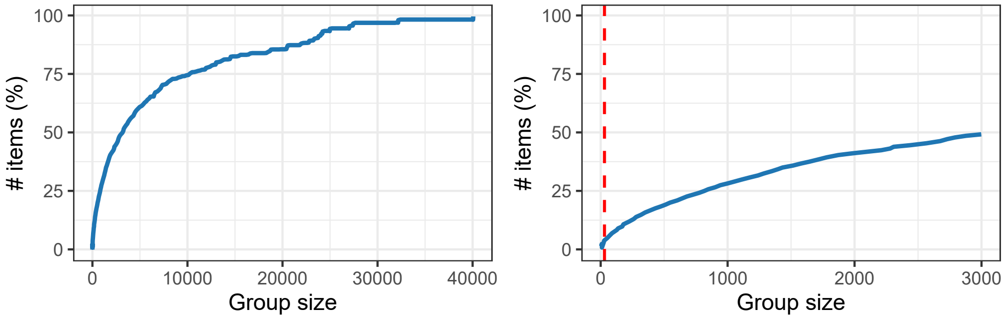

Figure 4: Cumulative Distribution of item counts in FT groups, with full distribution on the left and a zoomed-in view up to 3000 on the right. The red vertical line marks a group size of 30.

initial learning rate of 0.025 by default. SBERT was trained on Natural Language Inference (NLI) and was fine-tuned on a synthetic dataset generated from the million items in our real-world dataset. For E-SBERT, the weight factor σ was set to 1 after a thorough grid search.

To generate the synthetic dataset, we considered pairs of units of measure along with their conversion values and random item descriptions. In other words, for pairs of the same object, we generated descriptions with different measures that were equivalent in their conversions. We considered nine physical quantities: length, volume, mass, and area. In total, 100k items were generated, with 30k being duplicates.[6]

The synthetic dataset is created to develop document representations that effectively capture information about the scalar magnitudes of different objects. Consequently, the dataset was intentionally tailored to emphasize numerical terms and units of measurement, ensuring that the generated representations could effectively encode data related to these elements.

Finally, for the HDBSCAN algorithm, we carefully analyzed the parameter for the minimum size of clusters ($min\_cluster\_size$). After evaluating the results of the first token (FT) heuristic (see Figure 4), we noted that only 3.90% of the groups initially formed by token-based grouping had less than 31 items. Therefore, we set the minimum cluster size to 30 to refine the clustering process.

5.3. Evaluation Metrics

When evaluating the clustering performance, we faced the challenge of establishing a ground truth, a common challenge in unsupervised tasks. Hence we combined qualitative analysis with quantitative metrics commonly used in the clustering literature. In total, we consider four quantitative metrics that allow us to assess the coherence and structure of the resulting clusters:

Percentage of outliers. Identifies the percentage of instances that do not fit into any identified group. Outliers can indicate the presence of noise or ambiguity in the dataset. Therefore, a low percentage of outliers indicates good clustering quality.

Davies-Bouldin Score. Measures the average similarity between each cluster and its most similar cluster. The cluster similarity is a ratio between intra-cluster and inter-cluster distances, defined by the pairwise distance between centers belonging to different clusters. A lower score indicates better clustering, with values closer to 0 suggesting well-separated and distinct clusters.

Calinski and Harabasz Score. Assesses the intra-cluster and inter-cluster dispersion ratio. A higher score indicates better-defined clusters with a clear distinction between different groups.

Silhouette Coefficient. Measures the cohesion and separation of clusters. It is calculated for each instance and ranges from -1 to 1. A value close to 1 indicates that the instance is well-clustered and belongs to the correct cluster, while a negative value suggests that the instance may have been assigned to the wrong cluster.

5.4. Results and Discussion

We started our experimental analysis by running the FT heuristic, which provided a preliminary grouping of items. The distribution of group sizes resulting from this initial grouping is shown in Figure 4. As previously mentioned, the groups varied in size, with some containing as few as 31 items and others as many as 40,000.

Table 2 presents an overview of the clustering metrics for different model representations. The first two rows represent baseline results obtained without using the FT heuristic, while the subsequent rows use the FT+HDBSCAN approach. Here, we run HDBSCAN with two item representations (SBERT and E-SBERT) to check whether incorpo-

[6] Dataset available at: https://github.com/laic-ufmg/ESBERT/

Table 2: Evaluation of clusters found by clustering with different text representations.

Representation	*#Clusters*	*%Outlier*	*%Removed*	Calinski	Davies-bouldin	Silhouette
			HDBSCAN			
SBERT-NLI-synth	5,160	45.02	45.02	1.24×10^{6}	0.39	0.610
E-SBERT-NLI-synth	5,925	43.14	43.14	1.45×10^{6}	0.37	0.641
			FT+HDBSCAN			
Bag-of-Words	31,255	25.89	29.79	2,018.6	0.48	0.621
GloVe	33,565	22.62	26.52	1,986.6	0.475	0.628
fastText	33,278	20.13	24.03	2,295.4	0.45	0.643
SIF	36,086	23.58	27.48	2,730.4	0.415	0.679
Sent2vec	32,415	19.78	23.68	2,597.5	0.462	0.635
InferSent	32,572	27.59	31.49	1,742.8	0.48	0.618
SBERT-NLI	33,115	20.31	24.21	2,360.5	0.445	0.641
SBERT-NLI-synth	36,644	17.84	21.74	3,565.9	0.421	0.685
E-SBERT-NLI-synth	36,800	20.00	23.9	3,382.2	0.434	0.672

rating knowledge about item descriptions—as done by E-SBERT—enhances clustering outcomes. Additionally, S-BERT was fine-tuned only with NLI (suffix after the name of the method in the table) or with NLI plus the synthetic dataset.

Regarding the number of clusters generated, HDBSCAN with SBERT generated fewer clusters but a higher percentage of outliers when compared to other strategies. Despite the higher percentage of outliers (45.02%) than other methods, the silhouette coefficient results were significantly worse when compared to those obtained using the FT grouping. This suggests that applying a simple grouping strategy before clustering can improve the clustering quality.

Notably, the representation generated by E-SBERT, trained on the NLI dataset and fine-tuned on the Portuguese synthetic dataset, achieved the best results for the silhouette coefficient using Euclidean distance. It also yielded the best average results for the Calinski and Harabasz scores and the Davies-Bouldin scores (1.45×10^{6} and 0.37, respectively) when solely applied to HDBSCAN. We believe this is due to the high number of items categorized as outliers (43.14%), which are removed from the original collection of items for evaluating the clustering results.

E-SBERT did not significantly improve compared to SBERT when applied with FT. However, it achieved better results when used solely with HDBSCAN (second row of the table). Fine-tuning SBERT on the Portuguese synthetic dataset improved the clustering results for all metrics, emphasizing the importance of domain-specific fine-tuning for representation models used in unsupervised tasks such as text clustering.

Moreover, even though simple text representation strategies performed worse than SBERT, they yielded reasonable results. For instance, Bag-of-Words performed better than InferSent. However, our findings indicate that unsupervised methods such as SIF can yield results comparable to those obtained by SBERT, emphasizing the effectiveness of simple weighted average strategies for building sentence representations.

In conclusion, the experimental results suggest that using sophisticated supervised methods, such as InferSent, besides SBERT, to derive vectors for unsupervised tasks may not offer significant advantages. In contrast, unsupervised methods can provide results comparable to those of SBERT. Additionally, fine-tuning SBERT on a synthetic dataset tailored to the domain can significantly improve vector representations for item descriptions, as evidenced by the enhanced performance of clustering evaluation metrics.

Qualitative Analysis. To exemplify item groups obtained when applying HDBSCAN, using the representation obtained by SBERT trained on the NLI dataset and fine-tuned on the Portuguese synthetic dataset, Table 3 shows the largest subgroups for the first token “Mask”. The group “Mask”. The group “Mask” has 2,331 items. HDBSCAN split this group into 16 subgroups, each representing a specific variation of the item. Note that four subgroups, depicted in Table 3, have numbers as one of the most frequent tokens, highlighting the importance of numbers for describing items.

6. Conclusions and Future Work

In this work, we introduced a framework designed to handle non-standard text data characterized by diverse measures and numerical content. Our framework includes text preprocessing, extraction, and representation steps, offering a robust solution to handle such data. By evaluating various sentence representation methods, our framework enables the assessment of their efficacy in addressing the

Table 3: Subgroups of “Mask”.

Subgroup	# items	Most frequent tokens	Example of description
mask_1	490	"mask", "50", "elastic", "plastic", "triple", "with", "surgical", "box", "units"	plastic mask for surgical use box with 50 units
mask_6	279	"mask", "100", "plastic", "with", "elastic", "with", "polypropylene", "surgical", "unit", "layer"	plastic mask with elastic band and layer of polypropylene box 100 units
mask_0	270	"mask", "with", "for", "elastic", "reservoir", "plastic", "facial", "oxygen", "high", "reservoir"	mask for use in larynge for respiratory airway control
mask_11	234	"mask", "95", "n", "with", "filter", "elastic", "protection", "%"	mask for surgical use with protection n95 medium size against tuberculosis bacillus
mask_9	198	"mask", "3", "white", "with", "layer", "larynx", "for"	white mask for larynx made of faux fabric with 3 layers

challenge of grouping items, a task crucial for detecting overpricing and generating price statistics.

We proposed an enhanced approach to text representation, E-SBERT, focusing on capturing the essential components within sentences to produce more robust representations. Our experiments highlighted the effectiveness of combining simple heuristics, such as the first token grouping, with unsupervised text representation models like SIF and SBERT. These approaches outperformed more complex methods like InferSent. Furthermore, our findings underscored the importance of fine-tuning domain-specific data for unsupervised tasks, particularly in text clustering scenarios.

Limitations and Future Work. Despite the promising outcomes of our study, some limitations warrant attention for future investigations. First, establishing ground truth in our clustering task remains a significant challenge. While qualitative analysis was instrumental, correlating these qualitative insights with quantitative metrics poses challenges due to the limited number of analyses conducted. Additionally, the generalizability of our framework and methods across different domains and languages requires further exploration.

As future work, we can further analyze the clusters obtained by applying topic modeling techniques to gain deeper insights into their underlying themes. By extracting keywords using pre-trained document representation models, we can interpret the clusters as topics and use them for semantic search. Additionally, we plan to investigate the applicability of our text enhancement methodology in other contexts, such as product reviews and electronic health records (EHR).

7. Acknowledgements

This work was funded by the Prosecution Service of State of Minas Gerais (in Portuguese, *Ministério Público do Estado de Minas Gerais*, or simply MPMG) through its Analytical Capabilities Project and by CNPq, CAPES, FAPEMIG and the partnership project between AWS and CNPq.

8. Bibliographical References

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *ICLR*.

Junyang Chen, Zhiguo Gong, Wei Wang, Xiao Dong, Wei Wang, Weiwen Liu, Cong Wang, and Xian Chen. 2021. Inductive document representation learning for short text clustering. In *Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2020*, pages 600–616, Cham. Springer International Publishing.

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

Alexis Conneau, Holger Schwenk, Loïc Barrault, and Yann Lecun. 2017b. Very deep convolutional networks for text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1*, pages 1107–1116. ACL.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 4171–4186. ACL.

Balazs Godény. 2012. Rule based product name recognition and disambiguation. In *12th IEEE International Conference on Data Mining Workshops, ICDM*, pages 858–860. IEEE Computer Society.

Felix Hill, Kyunghyun Cho, and Anna Korhonen. 2016. Learning distributed representations of sentences from unlabelled data. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1367–1377, San Diego, California. ACL.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Vol 2*, pages 427–431. ACL.

Mansooreh Karami, Ahmadreza Mosallanezhad, Michelle V. Mancenido, and Huan Liu. 2022. "let's eat grandma": Does punctuation matter in sentence representation? In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD Proceedings, Part II*, volume 13714 of *Lecture Notes in Computer Science*, pages 588–604. Springer.

Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. context2vec: Learning generic context embedding with bidirectional lstm. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 51–61.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised learning of sentence embeddings using compositional n-gram features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 528–540.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014a. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. ACL.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014b. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992. ACL.

Yixiao Wang, Zied Bouraoui, Luis Espinosa Anke, and Steven Schockaert. 2022. Sentence selection strategies for distilling word embeddings from BERT. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference, LREC*, pages 2591–2600. European Language Resources Association.

Xiaoya Yin, Wu Zhang, Wenhao Zhu, Shuang Liu, and Tengjun Yao. 2020. Improving sentence representations via component focusing. *Applied Sciences*, 10(3).

Zheng Xin Yong and Tiago Timponi Torrent. 2020. Semi-supervised deep embedded clustering with anomaly detection for semantic frame induction. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC*, pages 3509–3519. European Language Resources Association.

Xikun Zhang, Deepak Ramachandran, Ian Tenney, Yanai Elazar, and Dan Roth. 2020. Do language embeddings capture scales? In *Findings of the Association for Computational Linguistics:*

EMNLP 2020, volume EMNLP 2020 of *Findings of ACL*, pages 4889–4896. ACL.