Comparing Different Methods for Assigning Portuguese Proverbs to News Headlines

Rui Mendes & Hugo Gonçalo Oliveira
CISUC, Department of Informatics Engineering
University of Coimbra, Portugal
rppm@student.dei.uc.pt, hroliv@dei.uc.pt

Abstract

This paper reports on the automatic selection of short-texts for amplifying the range of a given input's context, e.g. a news headline. Different methods are applied to select a corresponding expression from a list, which should be as semantically related to the input as possible. This study was developed for the Portuguese language, and considered expressions are proverbs, where figurative language is predominant. The set of explored methods includes some based on word overlap, others on static word embeddings, and also on recent contextual embeddings. To compare the explored methods in this subjective scenario, a survey was answered by humans, who rated the value of the selected expressions in terms of relatedness with regard to the input, and the humoristic value that may arise from this selection. The main conclusion was that simpler approaches, which end up selecting expressions that share words with the headline, are more easily related and considered to be funnier than other more elaborate approaches, which are more focused on the context.

Introduction

Topics involving the processing and understanding of natural language are often explored by applications designed for English. The study described in this paper targets the Portuguese language, which presents different challenges and, despite having a large number of speakers, has a much smaller research community. We propose an automatic selector of expressions, e.g. proverbs and sayings, able to choose expressions with regard to the input's context, e.g. a news headline. This type of expressions includes word-play and is rich in terms of figurative language. Thus, models trained in general language may struggle to interpret them. For this purpose, we test and analyze diverse approaches, and see many possible and different applications. In the domain of journalism, news stories are constantly published in online newspapers, raising the importance of having appealing headlines, which may be achieved by using familiar and figurative expressions that may also imply some form of humor. For instance, Jornal de Leiria's[1] headline *"Burro Velho não aprende línguas, mas mata a fome a quem aparecer"* ("Old donkey does not learn languages, but satisfies the customer's hunger") plays with the proverb *"Burro velho não aprende línguas"* ("Old donkey does not learn languages") and uses it to increase its appeal, as the news story is about a restaurant named *"Burro velho"* ("Old donkey"). One of the goals of this study is to automate that process of selection. There are also several tv shows, like the news satire *Last Week Tonight*, which use short-texts such as proverbs or movie titles to complement scenes. Furthermore, in the domain of chatbots, using proverbs and sayings in the appropriate contexts could make conversations more interesting.

However, for a computer, it is very troublesome to find the underlying meaning beneath the use of figurative language, which is not to be literally interpreted. In spite of that, a computer needs to understand how to find the most similar and humoristic relation between different texts, which in this study are represented by an input and a list of expressions. Several approaches can be used in the process of selecting the suitable sayings, starting by the representation of text and its comparison. The set of tested methods comprises simple approaches such as computing the Jaccard similarity or applying the Term Frequency - Inverse Document Frequency (TF-IDF) algorithm, to other more recent approaches such as those based on static Word Embeddings (WEs). Moreover, state-of-the-art methods based on transformers are also analyzed, namely the Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019), which has been getting a lot of attention due to having unexpectedly good results in many Natural Language Processing (NLP) benchmarks. Each of these approaches computes the similarity between two short texts, both with their advantages and disadvantages.

This paper is divided in four different sections, starting by this introduction, which is followed by an overview of background knowledge for better understanding this study. We further present some related works that inspired this paper, followed by the methodology by which this study was developed. Before concluding, we describe how the selected methods were evaluated in the proposed scenario and discuss our interpretation of the evaluation's results.

Background Knowledge

Before applying methods for natural language processing and understanding, a textual input often needs to be pre-

[1] https://www.jornaldeleiria.pt/

processed. Each sentence is often submitted to a morphological analysis consisting of: (i) tokenization, the separation of the sentence into words (tokens); (ii) lemmatization, reducing the word to its most basic form; (iii) part-of-speech (PoS) tagging, identifying the word's class (like nouns or adjectives). Therefore, each word and its derivatives are considered, eliminating words whose contribution to the semantic value of the sentence is limited, such as *stopwords*, which may be very frequent, e.g. *'the'* for the English language and *'a'* for the Portuguese language.

After pre-processing, the meaning of each token is considered, i.e. the semantic value of each word, bearing in mind that different words may have the same or a similar meaning (synonymy), e.g. *'big'* is semantically similar to *'large'*; or the same word might have more than one distinct meaning (homonymy), e.g. *'right'* may be a side or may mean *'correct'*.

Considering the semantic value of a word, the **distributional hypothesis** in linguistics, summarized by the quote *"You shall know a word by the company it keeps"* (Firth, 1957), assumes that the meaning of a word can be inferred by the context where it is inserted, i.e. the window of words that are near the chosen word.

The concept of **similarity** is grounded on features shared by two units, and sets their positions in a taxonomy. Semantic similarity measures the similarity of meanings, transmitted by words, and can be estimated by different methods, relying on different representations. A close concept is **relatedness** or association Budanitsky and Hirst (2006), which considers any other relation that may connect meanings. For instance, *dog* and *cat* are similar, but none is similar to *bone*. Moreover, *dog* is more related to *bone* than *cat*).

However, when it comes to longer sequences of text, the previous are less clear. In this context, **Semantic Textual Similarity** (STS) (Agirre et al., 2012; Cer et al., 2017) aims at computing the proximity of meaning of fragments of text or sentences. The most simple approaches for STS are based on averaging a semantic representation of each word Furthermore, it is common to weight each token according to its relevance, using, for instance, the TF-IDF algorithm for this purpose.

Traditional STS Methods

In the following paragraphs, we present the three simplest STS methods used in this study, considered traditional due to their usage before the introduction of word embeddings (WEs) and other more recent methods.

The **Jaccard** coefficient computes the similarity between sets as their intersection divided by their reunion. In order to compute sentence similarity using this measure, sentences are represented as sets of tokens. Similarity is then given by the number of shared tokens divided by the total number of distinct tokens in both sentences, possibly after preprocessing.

For the remaining methods, sentences are represented by vectors of numbers, and their similarity given by the cosine of such vectors, differing in how they are computed.

The **CountVectorizer** method (implementation in Pedregosa et al. (2011)) performs tokenization on a set of textual documents and constructs a vocabulary, enabling the codification of documents in regard to that vocabulary. These sparse vector elements (i.e., with many elements equal to zero) represent the number of times each word appears in the set of documents, and can be used to build co-occurrence matrices, which represent, for each word, the number of times it appears in the context of other words.

TFIDFVectorizer relies on the TF-IDF algorithm to compute the relevance of a word given a certain corpus, based on its frequency. It is usually used as a weighting factor in co-occurrence matrices. Term Frequency is the number of times a word w occurs in a document, while Inverse Document Frequency is the number of documents w appears in. This algorithm is able to reduce the weight of stop words, like prepositions or determiners, that contribute little to the meaning of the text, and increase the weight of words that do not appear very often elsewhere, and are thus more relevant for discriminating between different documents.

Static Word Embeddings

Representing words in vectors of real numbers, also called **word embeddings** (WEs), provides a friendly way of computing word similarity or using words as features in a machine learning framework, creating a semantic vector space. Such vectors are often learned from text, the more, the better, as they will better generalize word meanings. However, these models are limited to a single representation for each word, which means that multiple meanings of the same word are compressed into a single vector. WEs also present themselves as a solution to compress sparse vectors resulting from other techniques into dense vectors. They can also be improved by the utilization of the TF-IDF algorithm, as a way to increase the weight of the most relevant words of a sentence.

Regarding how WEs are learned, in this work, two different models were used:

- **GloVe** (Pennington, Socher, and Manning, 2014) is an unsupervised learning embedding algorithm based on a word co-occurrence matrix of probabilities in a textual corpus, finding relations between words.
- **FastText** (Bojanowski et al., 2017) is an algorithm for obtaining vector representations for words using a neural network, also considering character sequences, which may improve the processing of languages with a more complex morphology. It can use Continuous Bag-of-Words (CBOW), which uses the context to predict a word in its middle, or Skip-gram, which uses the distributed representation of a given word to predict the context.

BERT

A transformer (Vaswani et al., 2017) is a neural network architecture that converts input sequences in output sequences. Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) applies attention and recurrence mechanisms to gather information about the relevant context of a given word and then encode that context in a rich vector that smartly represents the word. Furthermore, a BERT model can be fine-tuned for a specific task, i.e. it can be specialized in the intended type of text, without having to be trained from scratch.

The **self-attention** algorithm in the transformer allows for the modelation of many downstream tasks, changing the adequate inputs and outputs. Each word of the input sequence has certain values due to its relations to other words, e.g. *"The dog ate its food."*, where *"its"* is related to *"the dog"*. This algorithm divides the input sequence, and, for each word *w*, calculates the scores of all words in relation to *w*. The output of the calculation is the sum of all returned vectors created in order to *w*, which is then passed as input for a feed-forward network. BERT has its own way of encoding tokens, starting with its WordPiece tokenization, which may even divide tokens into sub-tokens, e.g. *walking* or *walker* become *walk@@ ing* and *walk@@ er*. Even if the model does not know how to deal with the word *walking*, it probably does know other words that have *walk@@* in common, as it will appear more often. Because of this, it only needs the computation of the sentence vector, calculated through the average of the vectors of its tokens. It is also possible to use BERT to directly encode each sentence, instead of a token at a time, which may be an approach to be investigated in the future.

Related work

The development of automatic approaches for amplifying the range of a given story through creative artefacts is not new. In this context, systems have been proposed for generating poetry inspired by news stories (Colton, Goodwin, and Veale, 2012; Chrismartin and Manurung, 2015), metaphors according to the current news (Veale, Chen, and Li, 2017); new creative headlines, by resorting to figurative language (Alnajjar, Leppänen, and Toivonen, 2019), or blending them with well-known expressions (Gatti et al., 2015). Moreover, systems have been developed for simply recommending quotes to be used in dialogues (Ahn et al., 2016), without much concern regarding creativity.

Considering the generation of different types of text, namely poetry, Chrismartin and Manurung (2015) consider the dependency relations in a news story and use them for encoding the intended meaning of a poem. To produce text, they used a mechanism named chart generation, guided by the aforementioned relations.

Veale, Chen, and Li (2017) explored the generation of metaphors in regard to the current news. For selecting metaphors, considering their relation to the news, different techniques were explored, namely Latent Dirichlet Allocation (LDA), Latent Semantic Analysis (LSA) and Word2Vec. They were used for constructing a joint vector space that merged a news corpus with a metaphor corpus, thus facilitating their comparisons through the computation of the cosine similarity. After pairing headlines and metaphors, the authors crowd-sourced in order to evaluate the different pairing models in three dimensions: comprehensibility, aptness and influence of the metaphor on the reader's interpretation of the headline.

Alnajjar, Leppänen, and Toivonen (2019) adapted an automated journalism system that generates news headlines in English and Finnish to increase their creativity. In order to increase catchiness, headlines were enriched with the inclusion of a suitable known expression (e.g. movie title), or with the injection of figurative language, like similes and metaphors, according to the context of the news story. This was achieved with the computation of semantic similarity and by classifying the prosody of the phrase and the headline, which *"is evaluated to increase the catchiness of the result"*. For the news headline *"Biggest vote gains for The GreenLeague in Kuopio"*, the first approach generates the expression *"Alls well that ends well: Biggest vote gains for The Green League in Kuopio"*, while the figurative approach returned *"Biggest vote gains for The GreenLeague – the lovely god – in Kuopio"*.

Gatti et al. (2015) developed a method of generating catchy news headlines using well-known expressions that may also be used as a creativity boosting application. They start with sentences from a corpus of clichés, movie and song titles, or slogans. Then, for the creation of a sentence vector, they sum the sparse vectors representing the occurrences of the words of the sentence, excluding stop words. For each slogan, the most similar headlines were selected, according to the cosine similarity. Afterwards, the authors were able to classify the relations between words based on a dependency treebank, and select the keywords, from the selected news headlines, which were able to replace a word w in the slogan, considering its PoS. When successful, this substitution is ranked together with the other successful substitutions, considering the mean of its similarity and dependency scores. The candidate with the highest rank is selected and presented. Diverse outputs were returned. A example, for the headline *"Wood: Time for Wales to step up"*, based on the slogan *"Unleash the power of the majority"* is *"Unleash the power of the sun"*.

Ahn et al. (2016) proposed a system that recommends known quotes or expressions, given the features of the received short-line text, like dialogues and writing. In this work, there is a clear separation in the definition of context, between *pre-context*, i.e. texts before the quote, and *post-context*, i.e. texts next to the quote, within the capabilities of the selected *window*. This research presents five methods for quote recommendation:

1. Matching Granularity Adjustment: measures the importance of a set of contexts to a query.
2. Random Forest: tree based classification algorithm, chosen due to its resilience to overfitting and tendency to exhibit low variance and bias.
3. Convolutional Neural Network: searches for the best "*n-gram features in a given context by learning the parameters of fixed size filters for each n-gram*".
4. Recurrent Neural Network with LSTM: consists of three parts (forget, input, output) to teach the networks long-term dependencies without loss of information.
5. Rank Aggregation: groups the individual results of multiple methods to create a precise ranking of quotes.

Even though humour is not the main target of this work, some results may produce a humouristic effect. On the other hand, still in the scope of Computational Creativity, there are systems explicitly focused on the generation of humour, e.g. by replacing some words in a given text by others with the same form, context and topic, including taboo meanings (Valitutti et al., 2016), which are effective methods to increase the average funniness of a sentence.

Concerning the Portuguese language, related systems have focused on news headlines for the generation of memes, using a popular image macro and a text related to the news (Gonçalo Oliveira, Costa, and Pinto, 2016). Receiving as input a news headline, the previous system selects an image, from a predefined set, which is considered related to the input and adapts the text according to the stylistic rules that the meme textual content must abide, so that the combination of text and image produce humoristic content. Another example did not use news but Twitter trends, in this case as an inspiration for automatically generated poetry (Gonçalo Oliveira, 2017).

Methodology

We recall that the main goal of this work is to compare methods for assigning related Portuguese proverbs, automatically, to Portuguese news headlines. For this purpose, the main requirements are: (i) a collection of news headlines (in the future, these can be retrieved in real-time); (ii) a collection of expressions; (iii) an assignment method.

The development of this system was written in Python 3.6[2], with aid from various Python adapted libraries, both for text related operations, but for statistical purposes as well. The first step was to gather a good collection of data on the chosen context, accomplished by gathering news from the News API[3] sources, found in Portuguese newspapers' online editions[4]. The News API allows a client to get a maximum of one hundred current news on given keyword(s), returning an object with news whose titles are similar to the keyword(s). For this work, the news were reduced to their headline, in order to work with texts that do not differ too much in length, and were chosen both by date and by keyword, i.e. the API returned all the news related to the keywords *'clima'* ('climate'), *'ambiente'* ('environment') and *'aquecimento global'* ('global warming'), posted on the three months previous to the API call (February 2020).

Alongside the news, it is important to have a large enough corpus of proverbs. In this case, we used a corpus of 1,617 Portuguese proverbs, obtained from project Natura of Universidade do Minho[5]. Once we had the headlines and the proverbs, we decided on a range of methods for computing sentence similarity, to be later compared, namely:

- Jaccard Similarity
- TfIdfVectorizer
- CountVectorizer
- WEs generated by GloVe
- WEs generated by GloVe plus the weighing of the TF-IDF algorithm
- WEs generated by FastText (FT)
- WEs generated by FastText plus the weighing of the TF-IDF algorithm
- Bidirectional Encoder Representations from Transformers (BERT)

For both the **Count Vectorizer** and the **TF-IDF Vectorizer** method, Python's scikit-learn library (Pedregosa et al., 2011) was used to perform the calculations necessary to encode the words and their occurrences into matrices. These matrices are then used to compute the similarity between the input and the list of sayings, through the *cosine_similarity()* method from the mentioned library.

Four variations of **WEs**-based methods were tested. Using Python's Gensim library[6], two different models were used: Glove and FastText (using CBOW). Both were models pre-trained for Portuguese, with 300-dimension vectors: GloVe was obtained from the NILC repository of Portuguese word embeddings (Hartmann et al., 2017), and the FastText model from the FastText repository[7], where models are available for several languages, including Portuguese. Variations of sentence representations using the previous relied on TF-IDF for weighting word vectors. Moreover, to represent text with WEs, the short-text is submitted to a preprocess that includes tokenization and turning tokens to lowercase, accepting only those that are present in the model's vocabulary. The sentence vector is then computed from the average of its tokens' WEs. To finalize, we used Gensim's method for computing the cosine similarity – existent for any of the mentioned models – is used between the input's and proverb's sentence vector.

For the application of **BERT**, a pre-trained multilingual model was used, available by Google and covering 104 languages, including Portuguese: BERT-Base, Multilingual Cased[8]. For this method, a Python library named *bert-as-service*[9] was used, requiring to run a BERT server with the mentioned model, which is then called by the client, i.e. the system. Due to BERT's specific vector encoding, the cosine similarity is calculated with the help of Python's *NumPy*[10] library, instead of the previously mentioned methods.

For most of the methods, both headlines and proverbs were first pre-processed with the NLPyPort package (Ferreira, Gonçalo Oliveira, and Rodrigues, 2019), a layer on top of the Natural Language Toolkit (NLTK) (Loper and Bird, 2002) tackling Portuguese, specifically. This enabled the linguistic pre-processing, namely with tokenization and PoS tagging, and was essential for further application of the sim-

[2] https://www.python.org/

[3] https://newsapi.org/

[4] https://www.dn.pt; https://www.publico.pt; https://expresso.pt; https://www.sapo.pt; https://www.jn.pt

[5] https://natura.di.uminho.pt/wiki/doku.php

[6] https://radimrehurek.com/gensim/models/word2vec.html

[7] https://fasttext.cc/docs/en/crawl-vectors.html

[8] https://github.com/google-research/bert

[9] https://github.com/hanxiao/bert-as-service

[10] https://numpy.org/

ilarity methods, as seen above. Only BERT does not need this pre-process.

Afterwards, the similarity of each headline with each proverb is computed and the proverb with the highest similarity score is used. In this work, this is done for the eight tested methods, all relying in the computation of the cosine similarity between the vectors representing each sentence, which were computed by the average of the vectors of the sentence tokens. Following the computation of similarity, the proverb with the highest similarity score is selected to represent the correspondent approach, e.g. for the headline *"Produção de combustíveis fósseis cresce 50% acima do necessário para travar aquecimento global"* ("Fossil fuel production grows 50% above what is needed to curb global warming"), a good choice, in our opinion, would be *"Quem dá e torna a tirar ao inferno vai parar"* ("Those who give but take back, end up in hell").

Evaluation

Evaluating headline-proverb pairings is a subjective task. Therefore, in order to assess the results of each method, it was necessary to resort to human opinions, more precisely 24 volunteers who were asked to answer a survey. They were grouped into six teams of four people, each assigned to ten news headlines from a total of 60, summing a total of 240 different evaluations.

In the survey, created with Google Forms[11], for each of the ten assigned news headlines, the volunteer judge had to classify the selected expressions with regard to two aspects: (i) relatedness, which classifies the semantic proximity between the expression and the input; (ii) funniness, a classification for the humoristic value of the expression considering the input's context, but not limited by it, as a judge may find an expression funny by itself. Although we could think of other relevant aspects, we focused on the previous two. There was no need to validate the syntax (every expression was unaltered, only selected), and the aspect of originality is more pertinent for other endeavours, namely regarding works in text generation. For the scope of this evaluation, judges did not have to justify the score they gave each expression, as it could influence their opinion.

As an example, for the headline *"Emissões atmosféricas aumentaram em 2017"* ("Atmospheric emissions raised in 2017"), the volunteers were asked *"Como avaliaria a relação entre os provérbios e a notícia?"* ("How would you rate the relation between the proverbs and the news title?"). Below these questions, they would see the list of selected proverbs in a random order, with no repetitions, and rate each one according to a 4-point scale: Not related (1); Remotely related (2); Considerably related (3); Extremely related (4). Afterwards, regarding the funniness of each proverb, they were asked *"Relacionando com o título, quão engraçado é cada provérbio?"* ("In relation to the headline, how funny is each proverb?"), to which the answers were also rated on a 4-point scale: Not funny (1); Remotely funny (2); Considerably funny (3); Extremely funny (4).

Table 1 reveals the results of this evaluation, respectively for the relation between proverb and headline and its funniness. Besides the distribution of scores for each evaluated aspect, it shows the median (Md), the means (μ) and standard deviation (σ), which could work as an overall score. To measure inter-rater agreement, we used Fleiss' kappa (Fleiss, 1971), a coefficient similar to Cohen's Kappa (Landis and Koch, 1977), that also considers the possibility of judges agreeing by chance. The difference is that Cohen's Kappa only works for two judges, while Fleiss' applies to more than two. Since we had six different surveys, each answered by four judges, Fleiss' kappa was computed for each survey and each validated aspect. For relatedness, it was 0.094, and for funniness 0.054. Following the standard guidelines for interpreting these values, there is just a slight agreement in terms of relatedness and funniness, highlighting the subjectivity of this kind of evaluations.

As previously stated, the proverbs selected by each method were evaluated regarding their relatedness with the headline and whether they had humoristic value or not. Considering Table 1, the majority of the results were only satisfactory, both for relatedness and funniness. Most of the proverbs only had a slight relation to their headlines, with every method scoring at least 3 for more than 30% of the times. The idea that mixing proverbs with news headlines may give the headline a humorous touch is supported by similar results, with most methods scoring at least 3 in over 40% of the evaluations.

It is also possible to state that theoretically simpler methods achieved the best scores in the selection of proverbs for given headlines. Particularly, the simplest one, the Jaccard similarity, which is the only method whose proportion of top-scores (4) is higher than 20% in both realatedness and funniness. With regard to this realization, it is possible to argue that Jaccard has the highest scores due to its selection of proverbs with the most words in common with the headline, thus making it easier for people to make a quick and immediate connection between them. The same can be said for the TF-IDF and the Count Vectorizer, which follow Jaccard in the ranking of best relatedness scores.

A curious fact is that methods based in more recent semantic models, namely those using WEs, both with GloVe and FastText, and BERT, achieved lower scores. Even though they selected proverbs whose meaning may not be too far from the headline, their relation is, perhaps, not as pinpointed or clear as in the simpler methods. Considering the methods that rely on WEs, it is possible to conclude that those based on GloVe have higher scores than those using FastText, particularly in terms of funniness, where GloVe is the only method with over 40% answers scored with at least 3.

Table 2 is an indication of the methods with higher global success in terms of the number of times they were selected, by considering the proverbs which, based on the average of their four human opinions, scored at least 3.5 points in terms of relatedness. In the lead is again the simplest algorithm, Jaccard similarity, followed closely by TF-IDF Vectorizer.

The table also presents the average of shared tokens between the selected expression and its correspondent news

[11] https://www.google.com/forms/

Method	Relatedness						Funniness					
	$\mu \pm \sigma$	Md	1(%)	2(%)	3(%)	4(%)	$\mu \pm \sigma$	Md	1(%)	2(%)	3(%)	4(%)
Jaccard	2.4 ± 1.15	2	28.8	24.2	22.0	25.0	2.5 ± 1.06	2	22.0	28.8	27.5	21.7
Count Vectorizer	2.2 ± 1.06	2	35.1	26.3	24.6	14.0	2.3 ± 1.04	2	25.4	33.9	23.7	17.0
TFIDF Vectorizer	2.3 ± 1.09	2	32.5	24.6	25.9	17.0	2.3 ± 1.01	2	27.5	30.5	28.8	13.2
GloVe	2.2 ± 1.03	2	33.5	29.2	24.6	12.7	2.2 ± 1.02	2	32.6	23.7	33.1	10.6
GloVe+TFIDF	2.1 ± 1.05	2	37.7	26.3	23.7	12.3	2.2 ± 0.99	2	31.0	31.7	26.3	11.0
FT	2.0 ± 1.03	2	40.7	26.7	21.6	11.0	2.0 ± 0.97	2	39.4	28.4	25.0	7.2
FT+TFIDF	2.1 ± 1.03	2	39.0	25.4	25.0	10.6	2.2 ± 1.05	2	34.3	26.7	25.4	13.6
BERT	2.1 ± 1.04	2	41.1	23.7	24.6	10.6	2.2 ± 1.11	2	37.2	26.7	19.1	17.0

Table 1: Evaluation of proverb assignment to headlines.

Method	Top	μ(Intersection)$\pm\sigma$	
	μ(Rel)>3.5	Top	All
Jaccard	7	2.6 ± 1.3	2.0 ± 1.2
Count Vectorizer	3	3.7 ± 0.6	3.0 ± 1.5
TFIDF Vectorizer	6	2.3 ± 1.0	2.1 ± 1.3
GloVe	2	3.5 ± 2.1	2.2 ± 1.7
GloVe + TFIDF	1	2.0 ± 0.0	1.5 ± 1.4
FT	2	2.5 ± 0.7	1.7 ± 1.6
FT + TFIDF	1	3.0 ± 0.0	1.4 ± 1.1
BERT	1	3.0 ± 0.0	1.1 ± 1.1

Table 2: Number of selections with more than 3.5 average relatedness score (Top), and the average of token intersections between the proverb and the news headline, both for the top selections and for all selections by each method.

headline, for those top-related proverbs and also for all selected proverbs, for comparison purposes. This statistic is higher for the Count Vectorizer and Jaccard, two of the approaches with the most selections with at least 3.5 points of score in terms of relatedness. The more complex and recent approaches did not have as many successful selections, but even these selections had an average of at least two common words with the headline. However, when compared to the average of intersections for the total of selections, differences highlight that more related selections indeed share more tokens with the headline. Therefore, we may argue that people are quicker to relate two sentences that share many words, in regards to their full semantic value chosen by the approaches that are more up-to-date.

Despite having lower scores, BERT was able to make a selection that was scored with an average of 4 points. This was the third example in Table 3, where the chosen proverb's meaning and urgency clearly applies and is related to the headline. Moreover, BERT had one of the highest proportions of selections with maximum funniness. This might have been due to the surprising effect, or just by chance, as the means suggest.

Another good examples of a successful proverb selection, regarding the relation between title and proverb, is the first example in Table 3, which scored 4 for all its judges. Using the TF-IDF Vectorizer, the system selected a proverb whose meaning may correlate with the title's meaning, as they share the word *"lixo"* ("trash"), for example.

With concern to the best funniness related results, as seen in the fourth example of Table 3, the selected proverb had the average score of 4. It was selected by Jaccard similarity and uses taboo words, close to slang, which may be the reason for its high score. Taboo words "*are often used to produce humor effects*" (Valitutti et al., 2016).

In opposition to the highest-scored examples, two of the selections with the lowest score in terms of both relatedness and funniness are depicted in Table 4. Both of them were obtained with WEs, even though the first used the GloVe model and the second the FastText model. The first example tries to make use of the existence of the integer present in the headline, selecting an expression that allures to the commutative property of the multiplication of two real numbers, whose order does not change the end product. In the second example, it is difficult to grasp the scope of similarity between these two sentences, so much as to find their relation funny.

Conclusions

This study targeted a task of automatic text recommendation, in this case, Portuguese proverbs to news headlines. For this purpose, different semantic representation techniques were tested in this domain for computing the STS between proverbs in a corpus and headlines. To some extent, an application including some of those methods could be useful for writers and journalists, i.e., for making their news more appealing. The produced results were satisfactory, as most of the time people were able to establish a relation between the selected expression and the correspondent headline, and even often find it potentially funny.

Given some thought, the obtained results are particularly interesting, as they indicate that the proverbs sharing the same words with the headline, namely those chosen by simpler methods such as the Jaccard similarity, are more easily related to the headlines. On the other hand, proverbs selected by methods based on state-of-the-art models did not score as high amongst our volunteers. The former methods are based exclusively on the surface text, while the others rely on a deeper semantic representation of words. Though, regardless on whether they could select stronger related proverbs or not, according to our judges, they do not suit well this scenario, where figurative language is predominant. On this, Veale (2015) claims that human readers may suffer from a placebo effect, as they "*fill these containers with their own meanings, to see meaning in the outputs of generative systems where none was ever intended*".

	Result	Method	Rel	Fun
Headline	*”Malásia devolve 150 contentores ilegais de lixo a países subdesenvolvidos”* (”Malaysia returns 150 illegal trash containers to underdeveloped countries”)	TF-IDF	4	3.5
Proverb	*”Quem faz de si lixo, pisam-no as galinhas”* (”Whoever makes trash out of you, the chickens will stomp them”)			
Headline	*”Tempestade ’Glória’ fez 12 mortos em Espanha. Governo culpa alterações climáticas”* (”’Gloria’ storm made 12 casualities in Spain. Government blames climate change”)	TF-IDF	4	3.5
Proverb	*”A culpa morre solteira.”* (”Guilt dies single”)			
Headline	*“Ainda não é demasiado tarde para salvarmos os oceanos”* (”It is not too late to save the oceans”)	BERT	4	2.5
Proverb	*”Não deixe para amanhã o que você pode fazer hoje.”* (”Do not leave for tomorrow what you can do today”)			
Headline	*”Veredicto abre a porta a protecção para ’refugiados climáticos’”* (”Veredict opens door for protection to ’climate refugees’”)	Jaccard	2.5	4
Proverb	*”Para trás mija a burra.”* (”The donkey urinates backwards”)			
Headline	*”Judoca Jorge Fonseca galardoado com o prémio Ética no Desporto de 2019”* (”Judoka Jorge Fonseca is awarded with the prize ’Sport Ethics 2019’”)	Jaccard	1.5	4
Proverb	*”Não contes com o ovo no cu da galinha.”* (”Do not count on the egg being in the chicken’s butt.”)			

Table 3: Some expression selections whose relatedness between headline and saying, or funniness, was rated 4.

	Result	Method	Rel	Fun
Headline	*”Malásia devolve 150 contentores ilegais de lixo a países subdesenvolvidos”* (”Malaysia returns 150 illegal trash containers to underdeveloped countries”)	GloVe	1	1
Proverb	*”A ordem dos fatores não altera o produto.”* (”The order of factors does not change the end product”)			
Headline	*”Óculos de natação com realidade aumentada”* (”Swimming glasses with augmented reality”)	FastText	1	1
Proverb	*”Casa de pais, escola de filhos.”* (”Home of parents, school for sons”)			

Table 4: Some expression selections whose relatedness and funniness was rated 1.

In terms of future endeavours, a possible research direction would be to test supervised approaches for Semantic Textual Similarity (STS) in this scenario, including, but not limited to, fine-tuning BERT. Although there is no gold data with headline-proverb similarity available, we may try using collections for STS in Portuguese (Fonseca et al., 2016) for training such a model. We may also test the same approach with the recently available BERT model trained for Portuguese (Souza, Nogueira, and Lotufo, 2019), or try learning a model from Twitter or Reddit conversations where proverbs are used. This would require looking for tweets using any of the proverbs in the knowledge base and, if there is one, retrieve also the preceding publication(s).

This work was also key in the development of the creative system *TECo: Texto Em Contexto* (Mendes and Gonçalo Oliveira, 2020), in English *Text in Context*, which selects and adapts textual expressions based on a given textual input, also in Portuguese Adaptation methods result in new expressions, thus more novel, that still resemble original sayings, with increased relatedness, even when there are no related sayings available. The methods described here are responsible for selecting an initial set of expressions to adapt, and finally select the resulting expression to exhibit, out of several produced.

Acknowledgments

We to thank all the people who answered the survey, without whom the presented results would not be possible.

This work was partially supported by FCT’s INCoDe 2030 initiative, in the scope of the demonstration project AIA, “Apoio Inteligente a Empreendedores (*chatbots*)”.

References

Agirre, E.; Diab, M.; Cer, D.; and Gonzalez-Agirre, A. 2012. SemEval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of 1st Joint Conference on Lexical and Computational Semantics-Vol. 1: Proceedings of main conference and shared task, and Vol. 2: Proc. of Sixth International Workshop on Semantic Evaluation*, 385–393. Association for Computational Linguistics.

Ahn, Y.; Lee, H.; Jeon, H.; Ha, S.; and Lee, S.-g. 2016. Quote recommendation for dialogs and writings. In *CBRecSys@ RecSys*, 39–42.

Alnajjar, K.; Leppänen, L.; and Toivonen, H. 2019. No time like the present: Methods for generating colourful and factual multilingual news headlines. In *Proceedings of 10th International Conference on Computational Cre-*

ativity (ICCC), 258–265. Association for Computational Creativity.

Bojanowski, P.; Grave, E.; Joulin, A.; and Mikolov, T. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* 5:135–146.

Budanitsky, A., and Hirst, G. 2006. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational linguistics* 32(1):13–47.

Cer, D.; Diab, M.; Agirre, E.; Lopez-Gazpio, I.; and Specia, L. 2017. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 1–14. Association for Computational Linguistics.

Chrismartin, B., and Manurung, R. 2015. A chart generation system for topical meaningful metrical poetry. In *Proceedings of The 6th International Conference on Computational Creativity*, ICCC 2015, 308–314.

Colton, S.; Goodwin, J.; and Veale, T. 2012. Full FACE poetry generation. In *Proceedings of 3rd International Conference on Computational Creativity (ICCC)*, 95–102.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Procs. NAACL 2019*, NAACL-HLT 2019, 4171–4186. ACL.

Ferreira, J.; Gonçalo Oliveira, H.; and Rodrigues, R. 2019. Improving NLTK for processing Portuguese. In *Symposium on Languages, Applications and Technologies (SLATE 2019)*, volume 74 of *OASIcs*, 18:1–18:9. Schloss Dagstuhl.

Firth, J. 1957. *A Synopsis of Linguistic Theory, 1930-1955*. Blackwell Group.

Fleiss, J. L. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin* 76(5):378.

Fonseca, E.; Santos, L.; Criscuolo, M.; and Aluísio, S. 2016. Visão geral da avaliação de similaridade semântica e inferência textual. *Linguamática* 8(2):3–13.

Gatti, L.; Özbal, G.; Guerini, M.; Stock, O.; and Strapparava, C. 2015. Slogans are not forever: Adapting linguistic expressions to the news. In *Proceedings 24th International Joint Conference on Artificial Intelligence*, IJCAI 2015, 2452–2458. AAAI Press.

Gonçalo Oliveira, H.; Costa, D.; and Pinto, A. 2016. One does not simply produce funny memes! – explorations on the automatic generation of Internet humor. In *Proceedings of 7th International Conference on Computational Creativity*, ICCC 2016, 238–245.

Gonçalo Oliveira, H. 2017. O Poeta Artificial 2.0: Increasing meaningfulness in a poetry generation Twitter bot. In *Proceedings of the Workshop on Computational Creativity in Natural Language Generation (CC-NLG 2017)*, 11–20. Santiago de Compostela, Spain: ACL Press.

Hartmann, N. S.; Fonseca, E. R.; Shulby, C. D.; Treviso, M. V.; Rodrigues, J. S.; and Aluísio, S. M. 2017. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. In *Proc 11th Brazilian Symposium in Information and Human Language Technology*, STIL 2017.

Landis, J. R., and Koch, G. G. 1977. The measurement of observer agreement for categorical data. *Biometrics* 33(1):159–174.

Loper, E., and Bird, S. 2002. Nltk: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, 63–70.

Mendes, R., and Gonçalo Oliveira, H. 2020. Teco: Exploring word embeddings for text adaptation to a given context. In *Procs. of 11th ICCC*, Accepted for publication.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. 2011. Scikit-learn: Machine learning in python. *Journal of machine learning research* 12(Oct):2825–2830.

Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. Doha, Qatar: Association for Computational Linguistics.

Souza, F.; Nogueira, R.; and Lotufo, R. 2019. Portuguese named entity recognition using bert-crf. *arXiv preprint arXiv:1909.10649*.

Valitutti, A.; Doucet, A.; Toivanen, J. M.; and Toivonen, H. 2016. Computational generation and dissection of lexical replacement humor. *Natural Language Engineering* 22(5):727–749.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* 2017-December(Nips):5999–6009.

Veale, T.; Chen, H.; and Li, G. 2017. I read the news today, oh boy. In *International Conference on Distributed, Ambient, and Pervasive Interactions*, 696–709. Springer.

Veale, T. 2015. Game of tropes: Exploring the placebo effect in computational creativity. In *Proceedings of the 6th International Conference on Computational Creativity*, 78–85.