1 2 90

UNIVERSIDADE Đ
COIMBRA

Rui Pedro Pereira Mendes

TECo: Automatic Selection and Adaptation of Creative Text in Context

Dissertation proposal in the context of the Master in Informatics Engineering, Specialization in Intelligent Systems advised by Prof. Hugo Gonçalo Oliveira and presented to the Faculty of Sciences and Technology / Department of Informatics Engineering.

June 2020

Rui Pedro Pereira Mendes

TECo: Automatic Selection and Adaptation of Creative Text in Context

UNIVERSIDADE Đ
COIMBRA

1 2 90

Faculty of Sciences and Technology

Department of Informatics Engineering

TECo: Automatic Selection and Adaptation of Creative Text in Context

Rui Pedro Pereira Mendes

Dissertation proposal in the context of the Master in Informatics Engineering, Specialization in Intelligent Systems advised by Prof. Hugo Gonçalo Oliveira and presented to the Faculty of Sciences and Technology / Department of Informatics Engineering.

June 2020

1290 UNIVERSIDADE Ð COIMBRA

This page is intentionally left blank.

Acknowledgements

This master thesis is the last challenge of many years of effort, marking yet another step in this journey of mine. This process has been a long line of ups and downs, managing studies, having a sports career of high performance, and conciliating every other aspect of life. This would have been impossible to do without a base layer of support, for which I'm so thankful. There are people without whom this journey would have been harder, and possibly dispersed over a longer period of time. I would like to thank everyone who helped me in any way, particularly:

To my adviser from Department of Informatics Engineering of the University of Coimbra, Professor Hugo Gonçalo Oliveira, for being the most knowledgeable and available mentor a student could ask for. With each proposition and challenge, I grew as a researcher and expanded my areas of interest, for which I am very thankful.

To my Dad, for being my biggest counsellor and role model in any of the areas of my life. All the support along the course, all the efforts made in order for me to have a dual career, both academic and in badminton, make my successes every bit mine as yours. Thank you for the help, the patience, and the support, even when hardships emerged.

To my Mum, for being my biggest support. A simple, yet entirely true statement. The way you are able to provide solutions and always be available to help, even when you are not in your comfort zone, is truly remarkable. I would have write another thesis just to be able to thank you properly.

To Teresa, for being the coolest big sister one could ask for. As the master of linguistic minimalism, you always send the right message. I hope that I was also able to do it.

To my grandparents, for always looking at me with an unmeasured pride, sometimes probably exaggerated, that made me feel capable of achieving all of my goals.

To Scott and Leia, because there is not anything that a happy presence cannot fix.

To my group of close friends, that endured all the dramas and accepted every time I denied plans due to having a project or being in a tournament. Thank you for still liking me in the same way.

To Académica's badminton family, for being my quotidian companions for as long as I remember. Every conquest, in or out of the court, individually or as a team, belongs to everybody. I will always proudly carry our emblem in my chest.

To all my friends and colleagues, thank you for the hours of mutual help, and for the early nights of work and brotherhood. I hope to have contributed as much for your successes as you did for mine.

To every person that answered the surveys necessary for this work, a big thank you.

To everyone I did not mention that has supported me in any way, a heartfelt thank you.

Agradecimentos

Esta tese de mestrado é o último desafio de muitos anos de esforço, que marca mais um passo no meu percurso. Este processo tem sido uma longa linha de altos e baixos, tendo de gerir os estudos, uma carreira desportiva de elevado rendimento, e conciliar todos os restantes aspetos da vida. Teria sido impossível fazer isto sem uma base de apoio, pela qual estou tão agradecido. Existem pessoas sem as quais este percurso teria sido bem mais difícil, e provavelmente disperso por um maior período de tempo. Queria agradecer a todos os que me ajudaram, independentemente da forma, mas particularmente:

Ao meu orientador do Departamento de Engenharia Informática da Universidade de Coimbra, Professor Hugo Gonçalo Oliveira, por ser o mentor com mais conhecimento e a maior disponibilidade que um estudante poderia pedir. Em cada proposta e cada desafio, eu cresci como investigador e expandi as minhas áreas de interesse, pelo qual estou extremamente agradecido.

Ao meu Pai, por ser o meu maior conselheiro e modelo em qualquer uma das áreas da minha vida. Todo o apoio dado ao longo do curso, todos os esforços feitos em prol de eu poder usufruir de uma carreira dual, académica e desportiva, fazem dos meus sucessos tanto meus como teus. Obrigado por toda a ajuda, paciência, e apoio, mesmo quando os momentos mais difíceis surgiram.

À minha Mãe, por ser o meu maior apoio. Uma afirmação simples, e ainda assim completamente verdadeira. A maneira como consegues propor soluções e ter sempre disponibilidade para ajudar, mesmo em temas fora da tua zona de conforto, é inspiradora. Eu teria de escrever outra tese só para conseguir ser capaz de te agradecer devidamente.

À Teresa, por ser a irmã mais velha mais fixe que eu podia pedir. A mestre do minimalismo linguístico, consegue sempre transmitir a mensagem certa. Espero que eu também o tenha conseguido fazer.

Aos meus avós, por olharem para mim sempre com um orgulho desmedido, às vezes possivelmente exagerado, mas que me fez sentir capaz de alcançar todos os meus objetivos.

Ao Scott e à Leia, porque não há nada que não seja consertado com uma presença feliz.

Ao meu grupo de amigos mais próximo, que aturaram todos os dramas e aceitaram todas as vezes que eu disse que não podia fazer alguma coisa por ter um projecto ou um torneio. Obrigado por continuarem a gostar de mim do mesmo modo.

À família badmintonista da Académica, por serem uma companhia quotidiana desde que me lembro. Todas as conquistas, dentro e fora de campo, individuais ou de equipas, são de todos. Sempre carregarei o losango ao peito com o maior orgulho do mundo.

A todos os meus amigos e colegas, obrigado pelas horas de ajuda mútua e pelas noites madrugadoras de trabalho e convívio. Espero ter contribuído tanto para o vosso sucesso como vocês para o meu.

A todos os que responderam aos inquéritos necessários para este trabalho, um grande obrigado.

A todos que eu não mencionei, mas que de alguma forma me ajudaram, um sentido obrigado.

This page is intentionally left blank.

Abstract

Within Artificial Intelligence, the cluster of Computational Creativity has been receiving more attention as of late. In addition, its sub-field of Linguistic Creativity is trending with novel solutions for language generation, also supported by recent advances in Natural Language Processing. The work of this thesis is in the scope of the previous areas. It studies the development of a system that, given a short-text, e.g. a news headline, analyses its context for further utilization. Subsequently, the system aimed to achieve success in two different tasks: (i) the selection of an expression, e.g. a proverb or a movie title, which is most adequate for the given input's context; (ii) the generation of an expression, through adaptation methods, in such a way that it is semantically related to the input's context. As most of current systems dedicate their focus mainly to the English language, the system developed in the scope of this thesis, named Texto Em Contexto (TECo) (Text in Context), is dedicated entirely to the Portuguese language, and, more specifically, to the usage of figurative language.

Considering the first task, several approaches were explored, from simpler metrics such as the Jaccard coefficient, to the popular and traditional Word Embeddings, to state-of-the-art contextual embeddings based on Transformers, aiming to draw conclusions regarding the quality of the selected proverbs.

The second challenge required the research and development of new methodologies to adapt text based on a context. Three new methods were designed, implemented and tested, in order to draw conclusions on the different challenges proposed by the Portuguese language.

The obtained results show that the generated expressions were successful in terms of approximating to the input's context, producing novel and original content, despite some improvements that should be implemented in future endeavours, particularly concerning humour.

Keywords

Computational Creativity, Artificial Intelligence, Linguistic Creativity, Natural Language Processing, Portuguese language, Word Embeddings, Semantic Similarity, Natural Language Generation

Resumo

Dentro da Inteligência Artificial, a área da Criatividade Computacional tem recebido cada vez mais atenção. Dentro desta, a sub-área Criatividade Linguística está na moda, visto que se têm revelado novas soluções para a geração de linguagem, também suportada por avanços recentes na área do Processamento de Linguagem Natural. O trabalho descrito nesta tese situa-se no âmbito das áreas já mencionadas. Desenvolveu um sistema que, dado um texto curto como *input*, e.g. um título de notícia, é capaz de analisar o seu contexto. Subsequentemente, o sistema tem duas tarefas principais: (i) seleccionar uma expressão, e.g. um provérbio ou um título de filme, de modo a que seja a mais adequada para o contexto do *input*; (ii) gerar uma nova expressão através de métodos de adaptação, de modo a aproximar a expressão ao contexto do *input*. A maior parte dos sistemas actuais focam-se maioritariamente na língua Inglesa, e o sistema desenvolvido no âmbito desta tese, chamado Texto Em Contexto (TECo), dedica-se inteiramente à língua Portuguesa e, mais especificamente, ao uso de linguagem figurativa.

Considerando a primeira tarefa, diversas abordagens foram exploradas, desde métricas mais simples como o coeficiente de Jaccard, até representações mais populares e tradicionais como Word Embeddings, e ainda representações contextuais do estado-da-arte baseadas em *Transformers*, almejando tirar conclusões em relação à qualidade das expressões seleccionadas.

A segunda tarefa requereu uma investigação e desenvolvimento de novas metodologias para adaptar texto baseado no contexto do *input*. Três novos métodos foram criados, implementados, e testados, de modo a tirar conclusões acerca dos diferentes desafios propostos pela língua Portuguesa.

Os resultados obtidos mostram que as expressões geradas tiveram sucesso em termos de se aproximarem do contexto do texto dado, produzindo conteúdo novo e original, ainda que existam melhorias a implementar no futuro, particularmente a nível de humor.

Palavras-Chave

Criatividade Computacional, Inteligência Artificial, Criatividade Linguística, Processamento de Linguagem Natural, Língua Portuguesa, Word Embeddings, Similaridade Semântica, Geração de Linguagem Natural

This page is intentionally left blank.

Contents

This page is intentionally left blank.

Acronyms

AI Artificial Intelligence. 1, 11, 25

BERT Bidirectional Encoder Representations from Transformers. 4, 18, 20, 31, 32, 36, 44–46, 50, 53–56, 59–61

CBOW Continuous Bag-of-words. 15–17, 32

CC Computational Creativity. 1, 2, 5, 8–11, 20, 21, 25, 26, 60

CS Cosine Similarity. xiii, 17, 18, 20–23, 27, 30, 32, 33, 39, 41, 43, 45, 60

DEI Department of Informatics Engineering. 51

ICCC International Conference on Computational Creativity. 5, 6, 37, 44, 45, 48, 60, 68, 77

LC Linguistic Creativity. 1, 2

NER Named Entity Recognition. 14, 18, 20

NLP Natural Language Processing. 1, 2, 4, 5, 8, 11, 12, 17, 20, 31, 60

NLTK Natural Language Toolkit. 32

NLU Natural Language Understanding. 2, 3

PoS part of speech. 12, 13, 20, 23, 39, 40, 42, 45, 47

TECo Texto Em Contexto. vi, vii, xiii, 2, 5, 6, 20, 28, 30, 37, 45, 49, 50, 57–61

TF-IDF Term Frequency - Inverse Document Frequency. xiii, 4, 15, 18, 20, 22, 31, 32, 38, 44, 46, 50, 54–56, 58–60

WEs Word Embeddings. xiii, 4, 15, 17, 20–22, 27, 30–33, 36–38, 41, 43, 45, 48, 49, 60

List of Figures

List of Tables

This page is intentionally left blank.

Chapter 1

Introduction

Some say that computers have no intelligence and that the main difference between a human being and a computer is that the latter has not yet achieved its full potential regarding the creation of personal thoughts. Of course, if people were to believe in science fiction, they might fear creatively thinking machinery, as many artistic creations of this genre dictate a gain of conscience of the machines and, therefore, an uprising, usually against the human race. Henceforth, one of the hottest topics in engineering today is Artificial Intelligence (AI), even if, *"Success in creating AI would be the biggest event in human history. Unfortunately, it might also be the last, unless we learn how to avoid the risks."* [Hawking, 2014].

Amongst AI's trending areas, this thesis is included in the cluster of Computational Creativity (CC) and, more specifically, in the sub-field of Linguistic Creativity (LC), which, according to Colton and Wiggins [2012], aims to generate written artifacts that would be deemed creative, with the help of Natural Language Processing (NLP) tools and other linguistic resources. Throughout this thesis, it is possible to study and compare the details of various methods that were applied in the development of a system capable of both selecting and generating content adapted to a specific context, indirectly with some humoristic potential. The fact that different approaches, from traditional to state-of-the-art, may be used to achieve these goals, is also important to define strategies to draw comparisons between them. This enables findings regarding which are more suitable to both the selection and the generation / adaptation of text, as well as their main advantages and drawbacks. In addition, the work also leverages on the rich Portuguese culture and the development was directed for the Portuguese language.

In the remainder of this introductory chapter, Section 1.1 introduces some core concepts for this thesis, while the choice of this theme is thoroughly presented in Section 1.2, as well as the goals (Section 1.3), the used approaches through which to achieve them (Section 1.4), and contributions (Section 1.5) of this work. Finally, the structure of this document is discussed in Section 1.6.

1.1 Context

While it is obvious that computers have not yet achieved the full capability of communicating exclusively in natural language, there have been many attempts, such as chatbots or mobile phone personal assistants (e.g. Siri, Alexa,...). Even though they brought improvements to this area, they did not fulfil the utopian goal of giving a computer the ability to understand natural language and to respond in the same manner without sounding too distant from what a human being would say. Furthermore, even though most of the NLP applications have been naturally designed for the English language, considered the most globally accepted language for international communication, this project targets the Portuguese language, which, despite a large number of speakers, has a much smaller research community, and thus presents different challenges.

As said before, this project works in the area of LC, which requires the use of NLP techniques, including the sub-area of Natural Language Understanding (NLU). The idea of mixing natural language with CC is not new, but its applications could be quite remarkable. CC can be defined as: *"The philosophy, science and engineering of computational systems which, by taking on particular responsibilities, exhibit behaviours that unbiased observers would deem to be creative"* [Colton and Wiggins, 2012]. This definition packs three different areas: philosophy, science and engineering, the latter being the one with more emphasis in this work.

Generating creative content is subjective, since creativity may have different meanings for different people. Moreover, the same output may be interpreted differently regarding its context, originality or even humour. Humour is one of the most subjective types of content, for there is a lot of diversity in form, content and context for any joke or funny set-up. From the multiple ways to generate textual content, this thesis will explore and propose new methods to adapt texts based on some given context.

The key word in this thesis is **context**, which is present in the title of the developed system, Texto Em Contexto (TECo), that translates to *"Text in Context"*.

1.2 Motivation

To propose a tool able to, without human intervention, automatically present topics in a novel and lighter way, is the main proposition in this thesis. In any society's culture, linguistic expressions such as proverbs are present on a daily basis and are useful in many occasions, such as common dialogues, as a way to put emphasis on a given situation or occurrence. To amplify the range of a given story, either real or made-up, authors commonly reuse expressions or sayings known by a general audience as a title or subtitle, sometimes also achieving a humorous effect. For instance, in the domain of journalism, certain expressions may contribute to more appealing, creative and memorable headlines, e.g. the headline of Jornal de Leiria *"Burro Velho não aprende línguas, mas mata a fome a quem aparecer"*[1] ("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). This play with words aimed to increase the number of readers. In the domain of chatbots, using proverbs and sayings in the appropriate contexts could make conversations more interesting.

On the domain of news satire shows, the use of expressions like proverbs or movie titles, through some adaptation, is common. For instance, in Ricardo Araújo Pereira's show *"Isto é Gozar com Quem Trabalha"* (This is Mocking Those Who Work), the comedian and his team design sketches with trending news topics and adapt titles in accordance. An example, taken from the 29th March 2020 edition of the show is: for the problem the COVID-19 pandemic being ignored by the government of Brazil, which led to health prevention measures taken by the drug gangs from Brazilian *favelas*, they adapted the

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

movie title *"Tropa de Elite"* (Elite Troop) to *"Droga de Elite"* (Elite Drug), alluring to the problem at hand. In this format of television programs, like daily shows, it is common to adapt expressions to the context of the topic they want to address. If the expression is related enough, it can be used directly, but it may also suffer minor adaptations, to become more related to the context and still resemble the original saying. Nonetheless, the process to produce quality content may take a long time, as it requires a vast knowledge about folk and pop culture, and talent to be able to play with words in order to make good associations. The automation of this process does not aim to copy the human way of thinking, and does not guarantee better results, but provides faster suggestions that may be reviewed and even used by a human.

Creating content that Portuguese speaking people could relate to is ideal, as they are this project's target audience, and it is also the author's mother tongue. This motivates the use of Portuguese proverbs and sayings as a way to put emphasis or make an analogy on a given situation or occurrence, usually implying humoristic value to the situation. Most of these culturally enriched sentences already carry a lot of humoristic connotation by themselves, such as *"Burro velho não aprende línguas"* (Old donkeys cannot learn new languages), used when older people are not keen on learning or having new experiences outside the scope of their previous knowledge.

However, for a computer, it is very troublesome to find the underlying meaning beneath the use of these expressions, since they generally resort to figurative language and are thus 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 thesis are represented by an input, i.e. a news headline, and a list of proverbs and sayings with which to compare the input. The main Natural Language Understanding (NLU) techniques are not usually prepared or tuned to deal with the figurative language expressed in these sayings.

Several approaches can be used in the process of selecting the suitable sayings and in the process of adapting and generating new expressions, starting by the representation of text and its comparison. This thesis studies these approaches, both for selection and generation, in a methodical way as to discuss their advantages and setbacks with concern to the Portuguese language.

1.3 Goals

The proposed system aims to foment the interest in any topics it receives as input, namely news headlines, through a creative association and possible adaptation of known expressions. One hypothesis to draw more attention to certain topics would be making the news more appealing and thus easy to remember, possibly through humour. The aim of this project is to create novel content that, through the usage of figurative language and semantical relations, may have an increased humoristic value while keeping its baseline of context with regard to the input.

When considering different approaches to retrieve context data regarding the semantics of the textual input, it is important to develop an environment able to produce a valid comparison between them. Furthermore, regarding the goal of adapting and generating new expressions based on the input's context, there is a need to develop new approaches for the Portuguese language since, as previously mentioned, it does not have a large research community and presents specific challenges.

This thesis has two main goals, which will preempt two sub-systems:

- Explore different methods for **selecting** suitable short texts given a certain context, also supplied in the form of a short text. As mentioned in Section 1.2, there are several approaches to be studied and compared, from traditional and simpler algorithms to state-of-the-art methods that have been getting very noticed as of late. The selected text should be related to the input's context, be creative – possibly using figurative language – and, ideally, have the potential to provoke laughter (as a consequence, because this will not be specifically tackled);

- Propose new methods for **generating** short-texts suitable for a given context, through the adaptation of known expressions to further connect them to the specific context that has been addressed before. The adapted text should be syntactically coherent, be related to the input's context, be creative/novel, and, ideally, have the potential to provoke laughter (again, as a consequence, not because it was intentionally designed for humour). This thesis proposes three different methods that exploit this kind of word representation for adapting selected text, so that relatedness to the context increases.

Furthermore, in order for the second sub-system to select the best expression of the many it generates, it needs the prior sub-system to select the best option considering the input. However, there is a large risk due to the subjectiveness of the quality of the results, as automatic measures are probably not enough. Therefore, it was necessary to resort to the opinion of several human judges from different backgrounds, in order to achieve a successful qualitative and quantitative comparison.

1.4 Approaches

Different approaches were tested and studied, both for the selection based on context, and for generating new expressions through adaptation considering context as well.

The selection set of tested methods comprises simple unsupervised approaches such as the computation of the Jaccard similarity or 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 analysed, namely the Bidirectional Encoder Representations from Transformers (BERT) [Devlin et al., 2018], which has been getting a lot of attention due to having achieved the state-of-the-art results in many NLP benchmarks. Each of these approaches computes the semantic similarity between two short texts, and are able to select the most similar to be used. These approaches may function as a baseline to Semantic Textual Similarity [Agirre et al., 2012], which has been tested multiple times, but not so much on a scenario as rich in terms of figurative language.

With regard to the adaptation of text based on context, the proposed methods rely on WEs to: (i) replace a word by one from or related to the context; (ii) promote an analogy through the substitution of two words by two analogously-related, one of which is from the input's context; (iii) replace two words related to the input's context by computing the similarity of their relation instead of their own single value, in such a way that their relation is preserved.

For both selecting and generating methods, validation was based on the opinion of several human judges. For the first sub-system, the judges had to evaluate the selected expressions with regard to: (i) *relatedness*, classifying the semantic relation between the expression and the input; (ii) *funniness*, a classification for the humoristic value of the expression. On the other hand, the generation sub-system was submitted to a preliminary evaluation to validate and identify potential problems, by two human judges. This preliminary evaluation validated the generated short-texts in terms of *syntax*, i.e. classify the grammatical and/or structural issues of the expression, in addition to the aspects relatedness and funniness. Finally, for the final evaluation of the outputs, several human judges were gathered to answer surveys, classifying each generated output regarding relatedness, funniness, and *originality*, i.e. validating the expression's novelty, both in terms of the expression by itself and its relation to the input.

1.5 Contributions

In consequence of the achievement of the previously referred goals, this thesis contributed, in addition to itself, a system, named TECo, that is able to select and produce content written in the Portuguese language. As mentioned before, the selection of texts is based on the computation of the input's semantics. On the other hand, regarding the generation of content, it results from the adaptation of selected expressions with concern to the input. Both selection and generation tasks are provided with different approaches to successfully solve them, which will be compared and discussed in the following chapters.

The method comparison for the computation of semantic similarity was made in a challenging environment, due to the predominance of figurative language, with unexpected outcomes. More than TECo, the main contribution are the different used methods, along with their systematic evaluation. TECo, using said methodology, allows different people (e.g. journalists), to experiment diverse methods for generating creative texts based on an input. The users can even expand the number of generated expressions for each input, as to make a personal selection themselves. These expressions may be used as titles, sub-titles, social network commentaries, or just as creative inspiration. The system's code will be available online.

Besides the contribution of this thesis by itself, a full scientific paper was accepted at the International Conference on Computational Creativity (ICCC), available in appendix 7, while a second one, a short-paper, available in appendix 7, was also submitted to ICCC, the most relevant international conference on CC, which will be held in Coimbra, in September 2020. The first paper describes and discusses the several methodologies used in this thesis [Mendes and Gonçalo Oliveira, 2020]. The second submission was a short-paper about the methods proposed for text adaptation.

1.6 Document Structure

This document is structured in seven different chapters, each with various sections within. The present chapter is an introduction to the document, followed by Chapter 2, which presents the background knowledge to cover all the necessary topics for the development of this work, including defining humour in Section 2.1, as well as discussing CC in Section 2.2. Still in the second chapter, some approaches and methods are presented, both for NLP in Section 2.3 and Distributional Semantics in Section 2.4.

The following Chapter 3 portrays some of the works whose influence is essential to the development of this thesis, such as text selection systems and creative text generation systems.

From there on, Chapter 4 presents the first half of the work, which is in regard to the short-text selector based on context, presenting its methodology in Section 4.1 and further implementation in Section 4.2, while its evaluation and further discussion are in Sections 4.3 and 4.4.

Chapter 5 describes the second half of the work, introducing the proposed methods for adapting short-texts based on context in Section 5.1. Moreover, in Section 5.2, the methodology utilized to develop and implement the system is presented. It is followed by a first and smaller preliminary validation of results that aimed to validate methods, identify and fix minor problems that could have appeared, described in Section 5.3.

Chapter 6 describes the final stages of the work developed in the scope of this thesis, the TECo system, which is composed of both the previously mentioned selector and the proposed methods for adapting short-texts based on context. It dwells on the system's architecture in Section 6.1, followed by the evaluation of the work's results in Section 6.2. Section 6.3 presents the Twitter bot *@TextoEmContexto*, that was used for data retrieval and to publish some of the system's results. A discussion resuming the chapter's content is presented in Section 6.4, aiming to analyse the quality of the produced work, as well as draw some conclusions on the proposed methods.

To sum the contents of this thesis, on Chapter 7 recaps on this thesis' content, discussing its goals and describing the approaches utilized to complete them. In addition, possible future endeavours and contributions, that may arise from the development of this thesis, are discussed. In addition, both of the papers written for ICCC are added in the appendices of this thesis.

This page is intentionally left blank.

Chapter 2

Background Knowledge

In this chapter, relevant concepts for this thesis will be discussed in detail, in order to facilitate the understanding of the project. Section 2.1 will introduce some theoretical notions about humour and how to classify it. In addition, Section 2.2 will define creativity in a similar fashion as the previous section, before presenting some of the works in the field of Computational Creativity (CC). Section 2.3 is about the field of Natural Language Processing (NLP), and the tasks that it proposes to solve and how. To finalize the chapter, Section 2.4 presents in-depth knowledge of computational linguistics approach to semantic similarity, from theoretical proposals to manually constructed systems, before diving into core concepts of vector semantics, such as the Word2Vec model and the algorithms it uses when comparing text similarities. Furthermore, it will discuss state-of-the-art concepts such as language models based on Transformers.

2.1 Definition of Humour

Humour has been present in human interactions for as long as we can remember, in as many ways as possible, and is a part of the quotidian of people of every age and culture. Some studies on the subject go back to ancient Greek times. However, there is no such thing as an unanimous definition of humour, for its subjectivity is dependent on the one experiencing it, i.e., on their *sense of humour*.

Being able to create content of humoristic value is an inevitably subjective task. In spite of some common ground, there are many different definitions of humour. According to Martin and Ford [2018], *"Humour is a broad, multifaceted term that represents anything that people say or do that others perceive as funny and tends to make them laugh"*. However, Valitutti et al. [2016] divide the definition of humour between subjective and objective. On one hand, *subjectivity* depends on the receiver's ability to appreciate and recognize humour, and is reflected on actions such as laughter or smiling. On the other hand, *objectivity* is considered to be the content's *"capacity to induce a humour response"*, which can be perceived differently by multiple individuals.

Humour foments many experiences, usually associated to positive sensations and good mood, being a useful tool to improve the mental state of people, as well as augmenting their cognitive capabilities. According to Stock and Strapparava [2005], humour can be a way to support creativity, opening people's minds to new ideas and points of view. In addition, memorization capabilities are increased as it is *"a common experience to connect in our memory some knowledge to a humorous remark or event"*.

For many years there have been attempts to classify humour, and from the many classifications of theories of humour. A tripartite classification [Raskin, 2008] was adopted:

- **Incongruity theories:** this classification goes back to Aristotle, and refers to a bridge of semantic contrast, incoherence, or inappropriateness between the set of expectations and what is perceived. In this incongruity, humour arises.
- **Hostility / Superiority theories:** going back to Plato, this classification claims that there is humour in a feeling of superiority over something, even if that means being aggressive to a target.
- **Release / Relief theories:** humour as a form of liberation from an individual's constraints or psychic energy.

Linguistic forms of humour have a tendency to go along with theories of incongruity in an attempt to pinpoint the source of funniness in humour. Tagnin [2005] says that there are four linguistic phenomena that contribute to the humoristic value: (i) homonymy, words with the same form but different meanings; (ii) homophony, words with different forms, but identical pronunciations; (iii) polyssemy, words that have several different meanings; (iv) paronymy, words with similar forms, but different lexical meanings.

According to Attardo [2008], texts of humoristic value can be classified into three types: (i) comical plot with a punchline; (ii) plot with a rupture in the narrative to insert humoristic references that are contextually coherent for the audience or that are about the author; (iii) central complication within the plot, where the consequences aim to provoke laughter.

2.2 Computational Creativity

Before diving into the concept of Computational Creativity (CC), one must first dive into the concept of creativity, which does not present itself as an easy task. The creative process implies identifying *"who is the creator, what is the output, who is the audience, expectations and whether the output is unexpected or meets some goal, what are the inputs, whether feedback is being contemplated"* [Gervás, 2009].

2.2.1 Definition of Creativity

Replicating, understanding, or enhancing human-level creativity through a computer, are all tasks targeted by people who research in this area. Nevertheless, it will never be unanimous, since the very notion of "creativity" is very debatable. Although most concur with Mumford [2003] when he declared that *"creativity involves the production of novel, useful products"*, some refuse the pure definition of creativity, like Veale et al. [2013], who believe that a rule-based definition of creativity may lead artists to find false comfort in thinking that the definition may be *"corralled into a productive rule for generating creativity"*.

Barron and Harrington [1981] divide the criterion of creativity used in large bodies of research into two main categories:

1. *"Creativity as socially recognized achievement in which there are novel products to which one can point as evidence."*

2. *"Creativity as an ability manifested by performance in critical trials, in which one individual can be compared with another on a precisely defined scale."*

Creativity, when applied to computational systems, draws from both categories, for a system needs to bring novelty to the table in order to be creative, but has to perform as well, to ensure quality in both the development and the achievement of the desired results.

2.2.2 Creative and autonomous systems

One of the most addressed problems in CC is to increment the responsibility of engineered software on the creation of artefacts and ideas. Traditionally, computational systems have a problem-solving approach, even automated intelligent tasks are formulated in order to serve as a solution to the problem at hand.

However, to Colton and Wiggins [2012], *"it seems inappropriate to describe the composition of a sonata, or the painting of a picture as a problem to be solved, and so techniques that essentially come down to optimisation or classification are inappropriate"*. So, to distinguish CC research, they propose a new paradigm: *"artefact generation"*, where the automation of an intelligent system can be considered as an opportunity to generate valuable cultural content.

In the field of CC, there are many types of systems, and even though this thesis focuses on the linguistics field of research, others have been explored and are worth mentioning. We will focus on systems related with images and music, besides linguistics, and projects which interpret music to generate images:

- Zoric and Gambäck [2019], propose the use of evolutionary algorithms to generate images from musical inputs, operating through its meta-data, rather than focusing on raw data. Their project focuses on shared features between figures and the corresponding music, while their type (emotional or artistic features) is chosen by the user.

- Kim et al. [2019] presented a project that analyses content in fine art paintings, using a deep learning algorithm. This project's motivation was to understand what objects or subjects are mostly seen in fine art painting, as well as elucidating some co-occurring patterns that connect this content in art, interpreting the relatedness amongst different pieces of art.

- Rodrigues et al. [2019] proposed that mixing perceptual and cognitive discoveries with multiple characteristics of images, e.g. shape or size, may add *"value to a non-verbal communication and computational generation of multi-modal associations"*.

- *Neural Drum Machine*, developed by Aouameur et al. [2019], was an interactive system for real-time synthesis of drum sounds, operating as a suggestion maker for producers, who instead of doing the unproductive work of browsing countless drum samples that are available on the internet, or reducing their sample spectrum to the producers' favourite group of samples, *"which could hamper creativity"*.

- Cunha et al. [2020] developed a system capable of making visual adaptations to flags of countries, based on trending topics gathered from news sources. These adaptations could be in regard to colour, which may carry different meanings, or to symbols that match the textual input.

- Žnidaršič et al. [2016] created a CC infrastructure to support text-based creative systems. Resuming, the idea was to build an infrastructure that helps to adapt and develop creative artefacts from different concepts, like poetry, images or narratives, blending them through text resources.

Text generation systems

Besides the previous domains, which cover mainly music and visual arts, there are systems that regard the linguistic field of arts, and creativity in general. This domain is the core of this thesis' work, and, depending on their output, such systems can be divided into subgroups like narrative, poetry, short-line texts (e.g. slogans) or humorous text. Slogans were dissected and adapted by Gatti et al. [2015], who aimed to generate memorably catchy news titles, mixing ordinary news titles with slogans. Another related work was developed by Stock and Strapparava [2005], whose objective was to be able to develop a system able to adapt a known acronym to a funnier version of itself, modifying the acronym's full display.

However, since this thesis focuses more on these last two types of text, this section presents examples of the first two subgroups, whereas the latter ones will be discussed in Chapter 3.

Loller-Andersen and Gambäck [2019] created a system that, given a visual input, such as an image, generates a poem that aims to satisfy rhythmical constraints imposed by the visual context of the input. Generating poetry is no easy task, since it has certain constraints regarding form and content. Using a pre-trained Convolutional Neural Network image classifier to find objects in the input and combining tree search with a Recurrent Neural Network trained on a data set of more than 200,000 song lyrics, the system predicts outcomes without using methods based on rules. Instead, the previous combination guarantees poems with the correct form, while searching for optimal paths with words that rhyme.

Unanimity is not found in literary studies when trying to define the best features that make a narrative valuable, which is a serious difficulty to address by a computer. Storytelling generation systems were analysed by Gervás [2019], according to whom, on this genre, *"acceptability is valued highly and novelty is assigned less importance"*. For the author, generating acceptable stories should be prioritized, as it could function as a filter metric, before chasing novelty, which can be done afterwards. His system aims to create a story's plot representation in order to produce a range of outputs that might be accepted as instances of stories.

2.3 Natural Language Processing

Within Artificial Intelligence (AI), the field of Natural Language Processing (NLP) serves as a bridge of communication between machines and humans, and retrieves or generates data that is either text or human speech, as defined by Jurafsky and Martin [2019]. It is present in day-to-day applications such as translators (e.g. Google Translate), word processors (e.g. Microsoft Word) or personal assistants (e.g. Siri).

While this is a simple way of describing the field, natural language's characteristics, such as ambiguity or variation, even if fairly easy for a human who's fluent in the language to understand, are difficult for a machine to recognize, understand or generate. For example, for a human it may be simple to recognize sarcasm due to the tone of voice of the person speaking. However, a computer may fail to detect it and take the words literally. The knowledge about natural language necessary for a computer to be capable of understanding it can be divided in:

- **Phonetics:** the pronunciation of words.
- **Morphology:** the rules by which words are formed, including: (i) their part of speech (PoS), i.e. if they are a noun, a verb or else; (ii) their inflections, i.e. if they are singular, plural, male or female form; (iii) their irregularities.
- **Syntax:** knowledge about the proper structure of a sentence, i.e. order of words and its implications.
- **Semantics:** knowledge about the meaning of words an sentences. Its analysis is detailed in Section 2.3.2 and Section 2.4.
- **Pragmatics:** The true goal of a sentence, beyond its meaning, e.g. *"Break a leg"* in many contexts means *"Good luck!"*.
- **Discourse:** The context of a conversation, more than the simple meaning of the sentence alone.

Natural language can be referred to as unstructured data, since it lacks the rules that machines are used to. Therefore, NLP applies algorithms that aim to transform the data into a machine-readable form and extract the principal data from it. In order to do this, the traditional approach follows a *NLP pipeline* that may go through the following stages:

1. Speech Recognition
2. Morphology Analysis
3. Semantic Analysis

2.3.1 Morphological analysis

Syntax is considered to be a group of rules under which a sentence is structured, i.e., the correct form of the word sequence that composes the sentence. Even though the rules are not always clear and vary from language to language, computer algorithms apply said rules in order to retrieve meaning from them.

Word segmentation

Consists of splitting continuous text into multiple cells, i.e. tokens, which usually are words. Also called *tokenization*, for the input *"Cá se fazem, cá se pagam"* (You get what you do/deserve), an array of tokens is returned: ["Cá","se","fazem","cá","se","pagam"], not including spaces or punctuation.

Lemmatization

Reduces a given word to its most basic form through a morphological analysis, e.g. *"studies"* becomes *"study"*. Linguistics is key for this method, as it requires access to a dictionary of the language to be able to do the morphological analysis.

One may also opt for using **stemming**, which cuts a given word's prefixes and/or suffixes in order to reduce it to its basic form, i.e. *stem*. Even though it's simpler to develop a stemmer, this approach has some limitations. For the example given for the lemmatization, *"studies"* becomes *"studi"*, which is a good representation of the limitations of stemming, because *"studi"* is not a real word.

Part of speech tagging

Also called grammatical tagging, marks each word in the text with its corresponding part of speech (PoS), based on the definition of the word and its neighbouring words. The most relevant tags belong to open classes, as they are the most important to understand a sentence's meaning. These tags, like nouns, verbs, adjectives or adverbs, differ from closed classes (e.g. articles, prepositions, pronouns), as it is possible to enumerate every word belonging to these classes, thus being its denomination.

A PoS tagger is an application able to determine the PoS of a given word within its context. This can also be done by considering a well-composed morphology lexicon of a language that lists all words and their respective PoS, gender and number. However, the tagger's usage is particularly useful concerning words whose morphology depend on their contexts, e.g. in the Portuguese language, *"para"* can be a conjuction ("in order to"), a preposition ("for") or a verb (imperative of the verb *"parir*, which means "to give birth"). Without a PoS tagger and using only a morphology lexicon, it would not be possible to determine which is the correct PoS for this word.

Stop-word Removal

Stop-words are very frequent words that do not contribute to the meaning of the text, e.g. *"the"* for the English language and *"a"* for the Portuguese language. The removal of this kind of words does not take much value from a sentence's meaning.

2.3.2 Semantic analysis

Purely morphological and syntactic analysis is not sufficient for a good natural language interpreter. Bearing in mind the word *big*, it has many synonyms, such as *large*, *huge* or *giant*, different words with almost the same meaning. At the same time, when considering a single lemma like *right*, it has a property called homonymy, i.e. has different meanings according to context, since it can refer to *right* as opposed to *left*, or to *right* as in *correct.* However, it differs from **syntactic ambiguity**, where the ways of interpreting a sentence are due to its structure, e.g. in the sentence *"One morning I shot an elephant in my pajamas. How he got in my pajamas, I don't know"*[1], it uses the structural ambiguity provided by the fact that it is not obvious if the elephant was wearing the author's pajamas or if it was the author who was in his pajamas.

Semantic analysis' set of techniques is very complex and has received a lot of attention recently because of some revolutionizing state-of-the-art techniques, which will be explained in section 2.4.

To study semantics is to study the meaning of a language, i.e. the denotation of the components of the language, which can be words, phrases or signs. Their meaning and interpretation, as well as the structure of the sentence that incorporates such components, is aimed to be understood by computer algorithms. The relationship between individual words needs the previously studied syntactic approach in order to facilitate the under-

[1] Groucho Marx in Paramount Pictures' Animal Crackers, 1939, at url:`https://www.youtube.com/watch?v=NfN_gcjGoJo`

standing of properties like synonymy, polysemy, homonymy, amongst others. It is also important to emphasize the importance of concepts like **connotation**, which in the area of semiotics is the idea a word invokes in addition to its primary meaning, or **collocation**, which can be defined as a group of words that are likely to occur together, e.g. *climate change*.

Many metrics used to calculate semantic similarities use manually constructed systems such as **Wordnet**, which is a lexical database of English that groups words into sets of cognitive synonyms, i.e. *synsets*, according to a unique concept. This tool links not only words, but the different aspects of meaning of each word, i.e. *word sense*, disambiguating words that are in close proximity within the network, as well as labeling their semantic relations.

Some of the techniques used in this type of analysis are:

- **Named Entity Recognition (NER):** identifying and categorizing parts of text, i.e. entities, into groups, such as *'People'*, *'Places'*, *'Values'*, amongst others.
- **Word sense disambiguation:** based on its context, gives a certain word meaning, like the previous example with the word *'right'* whose meaning comes from the context on which it is inserted.
- **Semantic role labeling / Shallow semantic parsing:** attribute a role to each word within the structure of a sentence. For the sentence: *"The Professor taught the class."*, *"The Professor"* is the agent, *"taught"* is the predicate, and *"The class"* is the recipient.
- **Relation extraction:** as induced by its name, aims to extract the semantic relationships from text, between at least two entities. In the sentence *"Nelson Évora was Olympic champion"*, a triple of (Nelson Évora, was, Olympic champion) would represent the relationship, as *"was"* relates *"Nelson Évora"* to being *"Olympic champion"* in the past.

2.4 Distributional Semantics

Starting on the previously debated notion of semantics, one can consider distributional semantics as a theory that quantifies semantic similarities between linguistic objects based on their distributional properties within a large corpus of data. The **distributional hypothesis** in linguistics can be summarized by the quote *"You shall know a word by the company it keeps"* [Firth, 1957]. In addition, according to Harris [1954], words that are used and occur in the same contexts tend to have similar meanings.

The collection of all instances of a word and its linguistic context in a corpus is a way to put the theory to the test. Assuming the chosen word *bottle*, to know its meaning, one has to know the contexts where it is inserted. Every time the word *bottle* appears in text, it retains a window of context, i.e. the words near the chosen word. Afterwards, it is a good measure to remove every common word that does not have a lot of impact, like the article *the*, or combine some statistical methods to define with precision which words are more important when it comes to define the chosen word's meaning. This collection, when added the number of co-occurrences between *bottle* and other words, builds a vector space with a enormous amount of dimensions (i.e. the number of distinct words in the data), which allows the calculation of similarities, as will be described in detail in section 2.4.2.

Nonetheless, there are some methods to reduce these vast and sparse vector space, like Word2Vec, GloVe, or FastText.

2.4.1 Vector Semantics

Following the distributional hypothesis, the meaning of a word can be embedded in a vector that may be learned, unsupervisedly, from the contexts where this word occurs, in a large collection of documents. Such vectors are often called Word Embeddings (WEs). Representing words as numeric vectors is also a friendly way of computing word similarity or using words as features in a machine learning framework. These representations can also be named distributional semantic model used in computational linguistics, or semantic vector space.

The larger the amount of text, the better the capacity of Word Embeddings (WEs) to capture the meaning of words, as well as the diversity of the words used. 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.

The learned vectors achieve very interesting results when capturing syntactic and semantic regularities in language. Word Embeddings (WEs) also present themselves as a solution to compress sparse vectors that may arise if one is to calculate the **term-document matrix**. In this matrix, each row is attributed to a word in the vocabulary and each column represents a document from the given set of documents, and each cell presents a counter of times that a certain word appeared in each document. In addition, it is worth mentioning the **co-occurrence matrix**, which measures the number of times each word appeared in the context, or nearby, another given word.

To increase the quality of these measures, it is possible to weight the terms in the vectors by the utilization of the Term Frequency - Inverse Document Frequency (TF-IDF) metric, explained in Section 2.4.2.

In 2013, Mikolov et al. [2013b], while working for Google, created a statistical tool to create WEs from a textual corpus, called **Word2Vec**. This allowed a different approach, named 3CosAdd, regarding the math on these learned vectors. For instance, the resulting model could analyse the mathematical knowledge behind the expression $king - man + woman$ and conclude that it would be a vector close to *queen*, represented in Figure 2.1. Such linguistic regularities, present in the word vector space, can be perceived after training the tool with a large data set. The number of dimensions used in vector space is between 100 and 1000, most commonly 300, since it has been concluded that using more dimensions than that consumes too much time and memory while presenting very similar results. These vectors are also dense, i.e. most of their values differ from zero, in opposition to simple co-occurrence vectors.

Other researchers have since analysed the algorithm and compared it with models architectures of representation such as the Continuous Bag-of-words (CBOW) or the continuous skip-gram, seen in Figure 2.2. Both architectures were also developed by Mikolov et al. [2013a].

Continuous Bag-of-words (CBOW) is a simplistic representation used when retrieving information, where the textual corpus is represented by a set of its words, unconcerned with grammar or sorting order, but keeping track of the number of occurrences, which can be used as a feature to train the classifier.

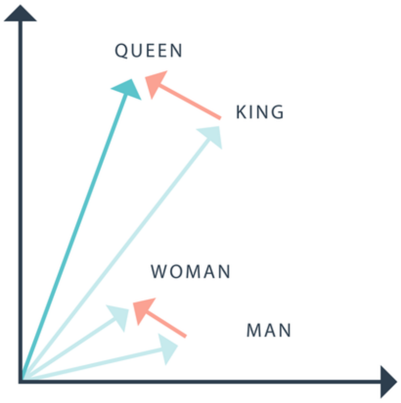


Figure 2.1: 3CosAdd approach example.

Models like **n-gram** are used to predict the next object in a sequential document. As most of the algorithms in this field, the larger the *n*, i.e. number of objects in the document, the better result the model offers, as it is able to save more data for each item. Alternatively, in computational linguistics, a different version of n-gram can be used, named **skip-gram**. In this case, the textual objects, usually words, do not have to be in a sequence, and may even have gaps between them. This approach puts emphasis on the context of the words, also called a *window* that englobes a determined number of words before and after the chosen word, whose weight is incremented when compared to other words. This window's size is usually 10 words for skip-gram.

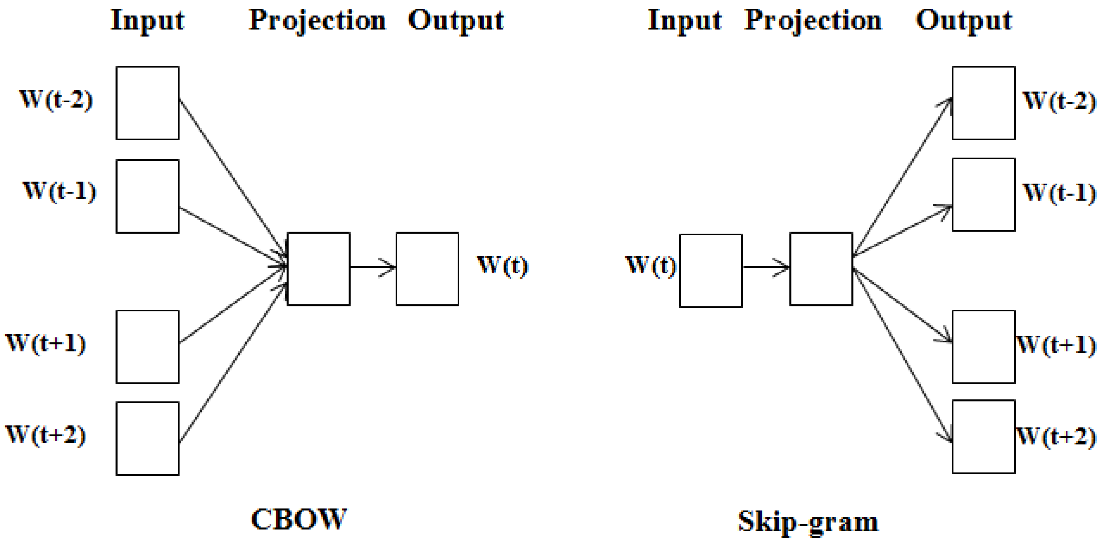


Figure 2.2: CBOW vs Skipgram

It is acknowledged that skip-gram is more efficient, particularly with irregularities, but CBOW is better in terms of speed.

To train the Word2Vec model, it is possible to choose hierarchical softmax and/or negative sampling. **Hierarchical softmax**, aiming to reduce the computational complexity, diminishes calculation by using a Huffman tree and handles well irregularities such as uncommon words. However, if the number of training epochs is too high, its efficiency decreases. On the other hand, **negative sampling** minimizes the percentage of weights it deals with, by choosing a small number of "negative" words, whose output should be zero. This behaviour allows this algorithm to handle frequent words, as well as vectors with a smaller number of dimensions.

A different example of an embedding algorithm is **Glove**, developed by Pennington et al. [2014], a method based on a word co-occurrence matrix of probabilities in a textual corpus, finding relations between words. Regardless of the embedding algorithm, both GloVe and Word2Vec models represent words as a compatible dense vector. Such representation is broadly used and, due to their compatibility, using one or another is often just a matter of changing the underlying model.

Another algorithm for generating WEs is **FastText** [Bojanowski et al., 2017], which obtains vector representations for words using a neural network to create WEs. This approach considers character sequences, which may improve the processing of languages with a more complex morphology. It can use 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.

2.4.2 Semantic Similarity

Measuring the similarity between words is one of the most important tasks when talking about semantics. A *rectangle* is not equal to a *pentagon*, but they are both geometric shapes and they are similar in that sense. Being able to compute the similarity is essential in computational linguistics. However, it is important to distinguish between similarity and **relatedness**, also called association, which includes any relationship between two words, but that does not mean that they're similar. A word may be related to another and yet be fundamentally different, e.g. *'cup'* and *'tea'*. It is not uncommon to have a cup of tea, but *'cup'* and *'tea'* do not share many similarities by themselves.

Cosine Similarity

Most measures for vector similarity in NLP use the cosine of the angle between vectors as their elected metric to compare words. The reason is that the inner product of two vectors, used in the calculation of the cosine, is as high as the values of the dimensions are large and coincidental between two vectors. However, according to Jurafsky and Martin [2019], the inner product has a major drawback in that it returns higher values the longer the vectors, meaning that more frequent words will have higher values than others. This is particularly undesirable when calculating similarities between two words. Hence, the proposed solution, seen in Figure 2.3 is to normalize the inner product by dividing by the length of both vectors to be compared, keeping the cosine angle between the two vectors.

$$\text{cosine}(\mathbf{v},\mathbf{w}) = \frac{\mathbf{v}\cdot\mathbf{w}}{|\mathbf{v}||\mathbf{w}|} = \frac{\sum_{i=1}^{N} v_i w_i}{\sqrt{\sum_{i=1}^{N} v_i^2}\sqrt{\sum_{i=1}^{N} w_i^2}}$$

Figure 2.3: Cosine Similarity (CS) formula.

Term Frequency - Inverse Document Frequency (TF-IDF) metric

TF stands for Term Frequency, which is the amount of occurrences of a word w in a document, while IDF stands for Inverse Document Frequency, with document frequency being the number of documents a word appears in. This algorithm, presented in Figure 2.4, is able to reduce the weight of stop words like prepositions and determiners, that contribute little to the meaning of the text, and increase the weight of words that do not appear very often and may carry more semantic value, and are thus more relevant for discriminating between different documents. These values, obtained by this transformation, compose vectors which may be used in the calculation of the Cosine Similarity (CS).

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

$tf_{i,j}$ = number of occurrences of i in j
df_i = number of documents containing i
N = total number of documents

Figure 2.4: Term Frequency - Inverse Document Frequency (TF-IDF) metric.

2.4.3 Transformers

In the field of Machine Learning, a transformer, proposed by Vaswani et al. [2017], is an architecture (Figure 2.5) that aims to provide a solution for converting input sequences into output sequences. It relies entirely on attention mechanisms to solve dependencies between input and output, avoiding the use of recurrence.

Bidirectional Encoder Representations from Transformers (BERT) [Devlin et al., 2018] is a state-of-the-art architecture that revolutionized the way people work with text data, handling sequence-to-sequence tasks and long-range dependencies. It 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. This model can be fine-tuned with a new output layer in order to be adapted for a broader range of tasks, i.e. to specialize in the intended type of text, without having to suffer the impact of considerable architecture changes.

Even though its goal is to predict the relations between text by analyzing them as a whole, Bidirectional Encoder Representations from Transformers (BERT) does not have the need to execute semantic analysis tasks such as Named Entity Recognition (NER). BERT's performance on tasks related to sentences and tokens is way more productive than many task-specific architectures.

As input, BERT receives a token sentence that may represent a single sentence or a pair of sentences, in the case of *Question - Answer* sets. The algorithm of **self-attention** in the transformer permits 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"*. It starts by dividing the input sequence in three vectors: query vector, key vector and value vector. For each word w, the mechanism will calculate 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. As it is computed several parallel times in a Transformer's architecture, some call this algorithm **Multi-head Attention**, represented in Figure 2.6.

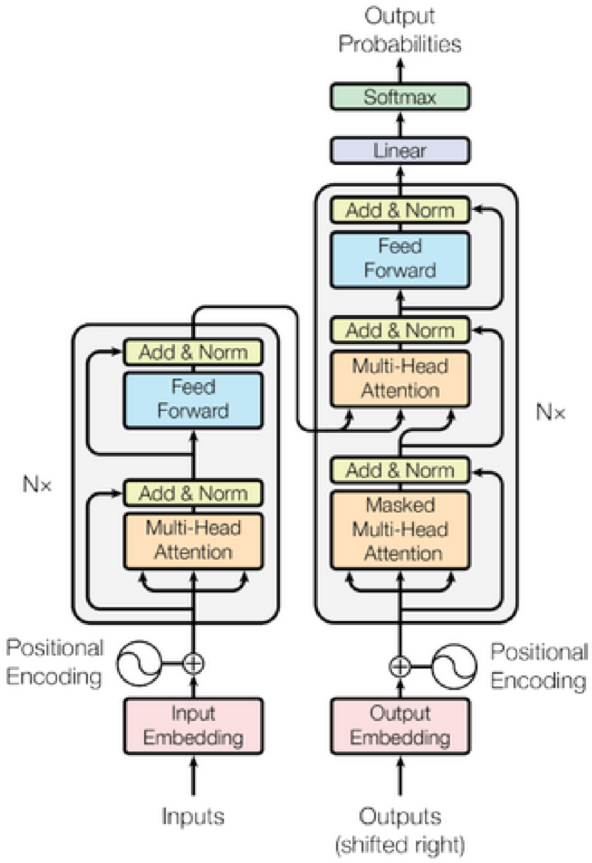


Figure 2.5: Transformer architecture.

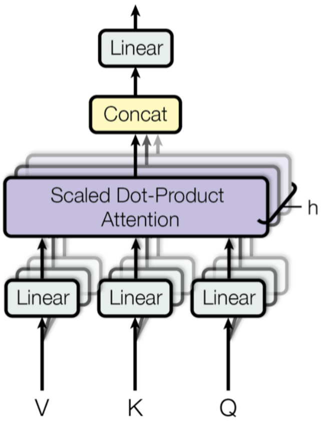


Figure 2.6: Multi-head Attention algorithm architecture.

Even if attention mechanisms provide an elegant solution, there are some disadvantages. They are only able to handle text strings with fixed-length, which means that the input may need to be split into smaller chunks before being used, possibly causing *context fragmentation*. Even though BERT can be fine-tuned for computing the similarity between sequences of text, we may also use it for encoding sentences in a vector, and then compute the cosine of such vectors.

2.5 Summary

To recapitulate, in this Chapter, some of the most important topics for this thesis were described. From the definition of humour in Section 2.1, the concept of CC was introduced in Section 2.2. Some creative systems were presented, including some in the scope of text generation. Section 2.3 describes some fundamental concepts in NLP, such as: (i) morphological analysis (word segmentation, lemmatization, PoS tagging and stop-word removal); (ii) semantic analysis (Named Entity Recognition (NER), word sense disambiguation). These morphological methods were often used throughout the development of the Texto Em Contexto (TECo) system.

The study of distributional semantics was discussed in Section 2.4, where the notion of linguistic context of a word was introduced by the distributional hypothesis. It was then demonstrated how vector semantics can be used for representing the meaning of a word through WEs, through techniques that may be applied with models such as Word2Vec, GloVe or FastText. For the computation of semantic similarity, the most common metric is Cosine Similarity (CS), whose efficiency may be amplified by the utilization of the Term Frequency - Inverse Document Frequency (TF-IDF) algorithm.

Finally, a reflection on transformers and, more closely, on BERT, was detailed, from its architecture to the attention mechanisms it uses.

Chapter 3

Related Work

This chapter discusses several works that relate closely to this thesis, in the field of Computational Creativity (CC), focusing more on linguistic related works and particularly in text related systems. As said before, this thesis proposes a system that handles short-line texts, like news titles, and produces creative text with potentially humoristic value, using Portuguese related proverbs. Therefore, some systems that handle semantic analysis, as well as the generation of new expressions, were given special attention.

During each work's analysis, four core areas will be explored: the *premise*, i.e. what the authors propose to achieve with the project, the *data* used as input, the *methodology* used to obtain the wanted results, and an *analysis* of the results and conclusions drawn from the experience.

On top of that, this chapter will divide the analysed systems across the axis of the type of textual format they work with. As previously stated, this thesis proposes a system divided in two sub-systems, on one hand to handle the selection of content related to a context and, on the other hand, to generate and adapt text based on its contexts, while also aiming to produce content with potential humoristic value. As such works related to the first sub-systems are described in Section 3.1, and those related to the generation of text are described in Section 3.2.

3.1 Text selection systems

Systems that receive short-texts as inputs and try to induce humour through their outputs are directly connected to this thesis' goal, and therefore must be debated in detail. As already mentioned, this thesis will focus on short-lined texts such as news titles and proverbs.

Ahn et al. [2016] proposed a system that would recommend quotes or expressions given the features of the received input, which would be short-line text such as 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*. One of the goals of this research was to present five new proposals for quote recommendation besides the more common Word Embeddings (WEs) approaches that calculated the CS between the query and the context of the quote.

1. **Matching Granularity Adjustment**: methods to measure the importance of a set of contexts to a query: (i) context clustering groups the set of contexts into

clusters that represent certain topics; (ii) context lumping compares the context of the quote to the query through CS with WEs plus Term Frequency - Inverse Document Frequency (TF-IDF) vector representation.

2. **Random Forest**: tree based classification algorithm, chosen due to its resilience to *overfitting* and tendency to exhibit low variance and bias. In this work's vision, the classifier was trained vectors of TF-IDF and their corresponding quotes.

3. **Convolutional Neural Network**: approach that searches for the best *"n-gram features in a given context by learning the parameters of fixed size filters for each n-gram"*. The model received as input the context in shape of a list of WEs vectors of the words in the context.

4. **Recurrent Neural Network**: uses a Long Short-Term Memory unit, consisting of three parts (forget, input, output) to teach the networks long-term dependencies without loss of information. Pre-trained WEs are used to map each word to word vector, and afterwards the data goes through fully connected layer and softmax layer as a way to calculate the probability of target quotes to reach recommendation status.

5. **Rank Aggregation**: with the idea of grouping the individual results of multiple algorithms to create a precise ranking. A candidate quote has a score inversely proportional to it's position in the ranked list of algorithms, plus the points of each of the quotes, to obtain a final score.

The authors concluded that the last method of the previous list, Rank Aggregation, produced results that outperformed the best baseline they had defined previous to their experiment, thus proving the effectiveness of grouping several algorithms to create a precise ranking.

Ameixa et al. [2014] developed a conversational agent, i.e. a chat-bot, that explores a corpus of movie subtitles to enrich the conversation with the user. The presented system selects an answer for a given input by taking into account the input and a large corpus of movie subtitles, computing the amount of interactions between pairs of sentences, e.g. the sentence *"So how old are you?"* triggered the answer *"That's none of your business"* a certain amount of times. At first, the system looks at triggers that are above a threshold of similarity to the input, computed with Jaccard coeffiecient and Overlap coefficient. This followed by the selection of the most frequent answer to that trigger, which also has to reach a similarity threshold in relation to the input. If no answers are selected, a random one is returned. For the input *"Are you joking?"*, the system selected *"Do I look like a joker?"*. The results were evaluated through a questionnaire, and the judges were asked to evaluate the expressions in a 5-point Likert Scale. Their concordance was computed with the Cronbach α measure.

3.2 Text generation systems

This thesis' second sub-system aims to adapt expressions in order to increase their relation to the input's context. A similar system was developed by Stock and Strapparava [2005] who tried to produce humoristic content in the form of acronyms. Inspired in the incongruity theory presented by Raskin [2008], this system aims to **analyse** acronyms and their full display of words, modify them and **generate** a new acronym, preferably with the same characters, but with different words in such a way that it would be funny.

This work's goal was to analyse the predefined criteria: (i) Simplify components of a larger system; (ii) Re-use standard and relevant linguistic resources, even if modifying and extending them; (iii) Choose and implement suitable humor related theories, i.e. developing algorithms capable of implementing them into a working prototype.

The system was divided into two sub-systems, the re-analyser and the generator. The **re-analyser** follows a defined schema of parsing the acronym, defining which words are good options for modification, e.g. adjectives, and which must not be changed. Afterwards, the system looks for substitutes for the words deemed to be modified through semantic fields, contemplating rhyme and rhythm, such is the importance of the new acronym sounding similar to the original. However, for words such as adjectives, they chose from antonym clusters obtained in *Wordnet*, aiming to produce irony and funniness when contrasting the original acronym. As an example, for the input *"FBI – Federal Bureau of Investigation"* the re-analyser produced the output *"FBI – Fantastic Bureau of Intimidation"*.

Concerning the acronym **generator**, the principal strategy is to consider as possible acronyms words that are ironic when related to the input, imposing constraints such as semantic reasoning or morpho-syntactic variations, aiming to produce some incongruity in the coherent sequence of these words. For the input where the main concept is *"writing"* and the attribute is *"creative"*, the system produced as output *"CAUSTIC – Creative Activity for Unconvincingly Sporadically Talkative Individualistic Commercials"*.

A group of 30 American university students were invited to evaluate 160 total products of the system, half from the re-analyser and the other half from the generator. They were rated along a scale of five levels of funniness, and the inquiry's results were positive, as 70% of the outputs were deemed to be at least *"funny enough"*.

Gatti et al. [2015] developed a project that covers generating short texts such as slogans for news titles, thus having a significant influence to this thesis's system. To the authors', this system's premise is creating a method of generating catchy news headlines using well known expressions, or as a creativity boosting application for artists. It relies on a method based on corpus of different types of text, such as cliches, movie and song titles, or slogans. To produce new expressions, they divided their system in three principal sections:

1. **Getting the news of the day** from a RSS feed, sorting it by date. Each news has a headline, a description and a link to the article.

2. **Extracting and expanding keywords**, i.e. important words in a text, defined by the probabilities calculated in a news corpus, from their number of appearances in headlines and the number of headlines in the corpus. Afterwards, pre-process them through tokenization and part of speech (PoS) tags, filtering stop-words and extracting their synonyms or related forms.

3. **Blending slogans with news** through lexical substitutions is a simple solution, with cohesion between neighbouring words being assured by PoS constraints and n-gram counts. However, the authors went further ahead and implemented a similarity check, using Cosine Similarity (CS), between the slogan and the news, to reduce the risk of producing outputs whose meaning was detached from what was expected, i.e. those that did not meet the similarity threshold.

   For each word (adjective, noun, adverb and verb), all words with whom it has a dependency relation were connected, with an estimation of how probable the words are to appear together. Afterwards, if the estimated probability is sufficient, the quality of a possible replacement of a word with the same PoS tag is ranked. Maximizing or determining a threshold for this ranking are ways to remove outputs with poor quality. Both the similarity and dependency scores are used to choose the best generated slogan.

Diverse outputs were returned, like one where the news headline is *"Wood: Time for Wales to step up"*, the output given by the re-analyser is *"Unleash the power of the majority"*, while generating the expression *"Unleash the power of the sun"*. The evaluation of these generated expressions was achieved by using Figure Eight[1], formerly known as CrowdFlower, a crowd-sourcing platform.

Generation of humour regarding short texts, often relates to word substitutions or modifications in order to increase the funniness of such text. One year later, Valitutti et al. [2016] defined three lexical constraints to perform this kind of replacements, considering a single word in each short text:

- **Form constraints:** aims to produce a *pun*, i.e. the use of a word that is similar to the original one, either orthographically or phonetically, with the intent of making a joke. This ambiguity augments the probability of producing funny content, due to its incongruity. *Orthographic similarity* is acceptable if it's possible to get a word with a single character deletion, addition or replacement. *Phonetic similarity* is measured by the orthographic similarity of the words' phonetic transcriptions. If a substitute word is orthographically and phonetically similar to the original and is either a noun, a verb, an adjective or an adverb, it's accepted by this constraint.
- **Semantic constraints:** the choice of words occurs according to their meaning or emotional connotations, which can be given by the theme of the text or by some prefixed topic. Taboos are a type of words whose connotation may produce humour in certain situations, for which the authors' define two categories: (i) *connotational taboo words*, where the taboo is in the pronounciation of the word; (ii) *taboo-inducing words*, i.e. words denoting taboo topics or animals when used as insults.
- **Contextual constraints:** take into account the position and textual context of the word to be replaced, such that the word forms a coherent and probable compound with its neighboring words in order to improving the semantic contrast of the substitution, thus making the text funnier. This contrast is measured through *n-grams* and the words must have a statistically relevant joint-probability in order to be selected.

The authors were able to draw some interesting conclusions from this research, using a corpus of SMSs as input and, replacing one word only, return an output with more potential humoristic value.

For validation, they sought responses from the crowdsourcing platform Crowdflower. Firstly, they evaluated each taboo word for its likability, profanity, discussibility and advisability, just to apply a **Cohen's Kappa coefficient** [Landis and Koch, 1977] This metric measures the inter-rater reliability between the responses and their own classification.

Afterwards, each subject had to evaluate the quality of a set of messages for funniness, enabling a comparison between the different constraints.

They concluded the large effectiveness of adding taboo words as a method to increase the average funniness of a sentence, while the other constraints' effects were not as noticeable, as they depend on word positioning. Furthermore, the effects of using constraints

[1] https://www.figure-eight.com/

on top of constraints was successful, as the overall rating of funniness increased with each constraint imposed.

Focusing on systems that are able to generate novel textual content from less than a sentence, it is possible to consider the system proposed by Gonçalo Oliveira and Rodrigues [2018] that handles riddles. To the authors, a riddle can be seen as a linguistic puzzle, its structure being divided into three steps: a question, a pause for the audience to think, and an answer, also called *punchline*. Generating a riddle is proposed as a three step process: Model Instantiation & Feature Acquisition, Riddle Creation and Rendering.

The first step, **Model Instantiation & Feature Acquisition**, defines the generation model, including the input of the *initial concept*, with two separate parts, either as a joint expression (e.g. *human rights*) or as a single word that can be split in half, each of the halves making each of the parts. Considering each part individually allows a subsequent retrieval of separate features from lexical resources such as Wordnet. In addition, grammatical information of a big corpus of word forms was obtained in a morphology lexicon named LABEL-Lex [Ranchhod et al., 1999] – also used in this thesis – to handle inflections and as source of words. Six different riddle generation models were then defined, experimenting with the two parts as a sum or as individual words, as well as creating new compounds with different words with similar phonetic values or with antonyms to create incongruence.

In **Riddle Creation** is fundamental that the connection between the concept and its features is understood in order to present in natural language. However, the knowledge base connects each word to a single meaning, thus missing some meanings for ambiguous words.

Rendering the riddle, i.e. transform the text into more fluent and natural language, was achieved following directives such as *"Noun features should appear before other features"*. This directive resulted in examples such as *"O que significa direitos humanos? um homem que não é está torto"* (What does human rights mean? a man that is not bent), as *"direito"* is Portuguese for "right" and for "plane", whose antonym is "bent".

As usual in works that return very subjective results, this work's generated riddles were deployed in a crowd-sourcing platform, as a form of validation. The audience subjects were not aware that the content they were evaluating was automatically produced, and were instructed to use a Likert scale for scoring values in the following aspects: interpretation, surprise, novelty and humour potential. The authors' found that most of their produced riddles were understandable, although sometimes only after a second reading. However, its novelty and surprise values were good, reflecting the accomplishment of one of their goals.

The work of Veale [2015] was a system that handles the generation of *tropes*, i.e. metaphorical usage of a word in the figurative sense. According to the authors, expressions like proverbs, particularly those that rhyme, give a perspective of deeper knowledge to the speaker than an ordinary expression with no rhyme. His system, a Twitter bot named *@MetaphorMagnet*, focuses on the generation of linguistic content such as tropes, implementing multiple rendering strategies to generate unexpected results from otherwise simple facts. Its outputs are generated by the system from its inputs that belong in *"its own knowledge-base of common-sense facts and beliefs"*. Furthermore, in this work human readers are submitted to a *placebo* effect, i.e. if they can recognize unintended meanings in the generated outputs. From Veale's perspective, this effect is present in all CC systems, benefiting *"from superficial users of surrealistic techniques to sophisticated knowledge-based Artificial Intelligence (AI) systems"*. It promotes itself as a consequence that the best short-line texts, e.g. *tweets*, come from an adequate, although non-obvious mix of common expressions with factual fillers.

The author defines four classes of approaches regarding metaphors, even though none offers a complete solution:

- **Categorical** approach views metaphors as a way to re-conceptualize one idea by placing it into another associated category.
- **Corrective** approach interprets metaphors as an anomaly and tries to retrieve the literal meaning. Needs literal case-frames on which deviant usages can be correctively projected.
- **Analogical** approach aims to capture the relation between the source and the mental representation it tries to achieve, through a graph representation of concepts to map sub-graph isomorphisms.
- **Schematic** approach tries to describe how metaphors represent deep seated cognitive structures, relying on a stock of Conceptual Metaphors to unearth the deep structures beneath the surface of diverse linguistic forms.

The *@MethaphorMagnet* aids the retrieval of Conceptual Metaphors through valuable public Web services, in order to produce novel linguistic metaphors and generate concepts that support its figurative tweets. Linguistic rendering enables the bot capability to group conflicts of ideas into a short-text narrative, just like a tweet. As example, for the tweet *"Remember when research was conducted by prestigious philosophers? Now, research is a fruit eaten only by lowly insects"*, the system produced the ironic output *"When some chefs prepare 'fresh' salads the way apothecaries prepare noxious poisons"*.

The authors assume that, for as much as a bot can represent humans, it cannot fully appreciate its content's value, which may be considered a key part of CC. In order to evaluate its generated tweets value, three metrics were used: *comprehensibility*, *novelty* and *retweetability* (the probability of an user considering re-tweeting the content). The crowd-sourcing users allowed the author to receive an idea of how meaningful *@MetaphorMagnet's* tweets are, but no pertinent conclusion on what that meaning is.

Veale et al. [2017] also explored the generation of metaphors in regard to the current news. The choice of metaphors, as well as their relation to the news, allowed the exploration of different techniques, such as Latent Dirichlet Allocation (LDA), an approach to topic modelling, the Latent Semantic Analysis (LSA), or Word2Vec. These techniques construct a joint vector space by merging the news corpus with the metaphor corpus, 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.

A multilingual system was studied by Alnajjar et al. [2019], who adapted an automated journalism system that generates news headlines in English and Finnish in order to increase their creativity. This system was complemented with existing tools for creative headlines, through the presentation of a suitable expression (e.g. movie title) as a catchy title, and figurative language generation, which injects similes and metaphors into headlines, according to the context of the news story, through methods of semantic similarity and 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"*.

Alnajjar and Toivonen [2020] developed a system to generate metaphorical slogans from a target concept and an adjective as property of the concept. Using multiple linguistic resources, from a repository of grammatical relations to semantic and language models plus genetic algorithms, the system is able to generate from the input *"computer; creative; artist;"* the slogan *"Talent, Skill And Support"*. For validation, the authors resorted to crowdsourcing judges to evaluate the slogans by: (i) relatedness; (ii) syntax; (iii) how metaphoric the slogan is; (iv) catchiness; (v) overall quality. Moreover, they also evaluated some manually created slogans by experts to have a baseline with which to compare, concluding that their metaphors were decent, producing at least one successful slogan for each input.

As the majority of systems of this kind, all the previous are for English, but we aim to work with Portuguese. In this 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 et al., 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]. Considering the generation poetry, the system developed by Chrismartin and Manurung [2015] tries to give meaning to poems, in addition to creating their structure, a challenge by itself. To produce text, they used a mechanism named chart generation, while capturing the semantic representations of news articles given as input from a dependency parser.

Returning to the generation of memes, Shimomoto et al. [2019] propose a method for automatically generating memes from a given news article. For any input and its context, there were two problems to solve: (i) Select an adequate image for the meme; (ii) Retrieve a short-text that can be used as a catchphrase. Similarly to the approach proposed in this thesis, the authors decided to represent texts through the usage of Word2Vec. Even the images were handled in this manner, as each image was represented by a set of tags (thus, represented by text), which were extracted using a deep neural network. For validation, the authors evaluated two aspects: (i) How related the meme was to the news article; (2) If the meme would be a more suitable choice over a randomly generated meme. On one hand, they concluded that the usage of memes might not be adequate in case the news article reports sad incidents, but on the other hand their framework's generated memes were preferred over randomly generated memes, exhibiting some positive remarks.

Bay et al. [2017] provide a text transformation computational process using WEs to approximate the semantic value of the texts and constraints to lead the word replacement. They applied this process to poems and song lyrics, so that the output would be text dictated by a human voice. The definition of primary operations used for text transformation was mandatory: (i) Similarity computed through the usage of CS, e.g. *walking* returns *running*; (ii) Neighbor, words that have similar usages or associations with the input word, e.g. from *apple* it returns *tree*; (iii) Theme, e.g. from *actress* it deducts *female* or *woman*; (iv) Analogy, using the logic that word *a* is to word *b* as word *c* is to word *d*, e.g. from *roof* and *castle*, it returns *parapet* and *battlements*. Afterwards, the authors defined the constraints, e.g. the word is *"found in an English dictionary"*, and defined rules for the transformations while also defining the intention of each transformation. They validated their process in an evaluation of quality, thematic accuracy and source identification or familiarity.

3.3 Discussion

The previously discussed papers were chosen due to the fact that it was possible to find several similarities between the described systems and this thesis' system. Most of the systems depicted in this chapter are summarized in Table 3.1, where it is easier to establish relations to this thesis' system, also summarized in the table. The majority of these related works receive as input short-texts, either in the format of news headlines, tweets or short messages (SMSs). Texto Em Contexto (TECo) is able to receive as input short-texts as well and, for the purpose of this thesis, news headlines were chosen.

There are different approaches, depending on the goals of the studies, but most of them end up mixing morphological and semantic analysis, as well as linguistic generation. Many of these methods served as inspiration to those described in this thesis, even if not directly applicable, like the chart generation or the image macro selection, which do not part in this work. However, focusing on the outputs, many were short-texts returned by their systems, sometimes with humoristic value in between, and as such are similar to this thesis proposed work. One has to consider that even memes need a valuable catchphrase, a short-text in its core.

Regarding these works methods for validation, most addressed the inherent subjectivity through human evaluators, either in person, by dealing with several invited evaluators, or by recurring to outer sources such as crowd-sourcing platforms.

References	**Input**	**Approach**	**Output**
Ahn et al. [2016]	Short-texts (Tweets and Quotes)	Various Approaches	Short-texts
Ameixa et al. [2014]	Short-texts (User input)	Semantic Selection	Short-texts
Gatti et al. [2015]	Short-texts (News headlines)	Lexical substitution	Short-texts (Slogans)
Valitutti et al. [2016]	Short-texts (SMSs)	Lexical substitution	Short-texts
Stock and Strapparava [2005]	Acronyms + Full display	Semantic substitution + Linguistic Generation	Acronyms + Full display
Gonçalo Oliveira and Rodrigues [2018]	Short-texts (Single word or joint expression)	Linguistic Generation	Riddle (Question + Answer)
Veale [2015]	Short-texts	Linguistic Generation	Short-texts (Metaphors)
Veale et al. [2017]	Short-texts (News Headlines)	Semantic Selection	Short-texts (Metaphors)
Alnajjar et al. [2019]	Short-texts (News Headlines)	Linguistic Adaptation	Short-texts
Alnajjar and Toivonen [2020]	Concepts (Three Words)	Linguistic Generation	Short-texts (Slogans/Metaphors)
Gonçalo Oliveira et al. [2016]	Short-texts (News Headlines)	Image Macro Selection + Semantic Selection	Memes
Gonçalo Oliveira [2017]	Set of Keywords (Twitter Trends)	Linguistic Generation	Long-texts (Poems)
Chrismartin and Manurung [2015]	Long-texts (News articles)	Chart Generation	Long-texts (Poems)
Shimomoto et al. [2019]	Long-texts (News articles)	Image Macro Selection + Semantic Selection	Memes
Bay et al. [2017]	Poems and Song Lyrics	Linguistic Generation	Poems and Song Lyrics
Mendes and Gonçalo Oliveira [2020]	Short-texts (News Headlines)	Semantic Selection & Linguistic Generation	Short-texts (Expressions)

Table 3.1: Comparison of the related works structures.

Chapter 4

Short-Text Selector based on Context

As previously mentioned, Texto Em Contexto (TECo) was designed to have two main goals: to select and to adapt textual expressions based on the context of the input, also a short-text, e.g. a news headline, like *"Nações Unidas alertam que temperatura do planeta vai aumentar"* ("United Nations warn that the planet's temperature will rise"). This chapter describes the first goal and the sub-system developed with that goal in mind. As such, it compares several methods for automatically assign expressions to an input – considering the Portuguese language – that are defined in Section 4.1.

For the purpose of evaluating this set of methods, an experiment was set up, both in terms of implementation and design of the evaluation. Its main requirements were: (i) a collection of texts to input, e.g. news headlines; (ii) a collection of expressions, e.g. proverbs and movie titles; (iii) an assignment method(s).

4.1 Selection Methods

The following step was to define a range of methods for computing the semantic similarity of sentences, to be later compared. Several methods were tested, based on different text representations, from the most traditional that consider only *surface text* – the words by themselves, with no regard for meaning – to representations based on static Word Embeddings (WEs) and contextual WEs, which can be said to be the state-of-the-art. The first step in any of these methods is the word segmentation, also named tokenization, representing a sentence in an array of words, also named tokens.

The first method is **Jaccard** coefficient, that computes the similarity between sets of tokens, i.e. words from the sentences to be compared, as the intersection of tokens between sets divided by their reunion. This formula is presented in Figure 4.1.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Figure 4.1: Jaccard coefficient's formula.

For the remaining methods, sentences are represented as vectors of numbers, and their similarity is given by the Cosine Similarity (CS) of such vectors, differing in how the vectors are computed. **Count Vectorizer** performs tokenization on a set of text documents and constructs a vocabulary, enabling the codification of documents in regard to that vocabulary. These sparse vectors (as they have many elements equal to zero) represent the number of times each word appears in the set of documents. **TFIDF Vectorizer** is based on the Term Frequency - Inverse Document Frequency (TF-IDF) algorithm, that aims to compute the relevance of a word in a document given its frequency in the document and in a large corpus. This method is commonly used as a weighting factor in co-occurrence matrices, which represent the amount of times, for each word, that other words appear in its context.

Furthermore, more methods were tested, including four based on static **WEs**, more precisely, in two different embedding algorithms, Glove and FastText. These algorithms represent words in dense vectors, and enable the retrieval of the sentence's vector representation, through the average of its tokens WEs. Both GloVe and FastText were validated with and without the application of the TF-IDF metric. Moreover, if the TF-IDF metric is chosen to be applied, the token's vector is multiplied by the TF-IDF value, thus increasing the value of the most relevant words in the sentence. WEs are a friendly way to represent words, as they are known for keeping linguistic regularities, including word similarity and other relations. They are a more sophisticated way to compress sparse vectors.

Finally, to cover all the different methodologies detailed in Chapter 2, BERT was also tested. As was said, BERT is an architecture that recently lead to state-of-the-art results in many Natural Language Processing (NLP) benchmarks. Through the application of attention and recurrence mechanisms, it gathers information about the context of tokens and encodes their meaning in contextualized vectors or sequences of tokens (e.g. sentences) in a single vector.

4.2 Experimental Setup

In this section, a more detailed explanation regarding the implementation of the methods presented in the previous section will be given. The development of this system was written in Python 3.6[1], with aid from various Python adapted libraries, both for text related operations, but for statistical purposes as well. To be given as input, 30 news headlines were retrieved from different sources. They were collected through the News API[2], which allows a client to get current news on given keywords, e.g. *"clima"* (climate) or *"ambiente"* (environment). The corpus of expressions to be selected was composed by 1,600 Portuguese proverbs and sayings from project Natura[3].

Every method presented in this section requires a pre-process, including the word segmentation operation, also known as tokenization, in order to represent the short-text as an array of words, known as tokens. With concern to the theoretically simpler algorithms, **Jaccard** computes sentence similarity using this measure, as 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, as seen in Figure 4.1.

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.

[1] `https://www.python.org/`

[2] `https://newsapi.org/`

[3] `https://natura.di.uminho.pt`

Four variations of methods were evaluated in terms of **WEs**. As far as the generation of the WEs is concerned, two different models were used: Glove and FastText (using Continuous Bag-of-words (CBOW)). Each used model was pre-trained for Portuguese, with 300-dimension vectors, with GloVe being available as NILC embeddings [Hartmann et al., 2017], and the FastText model[4] being multilingual (147 languages, including Portuguese). With concern to the weighing of the WEs, a solution through the utilization of the TF-IDF algorithm was used, as a way to improve the quality of the vectors. To represent text with WEs, the short-text is submitted to a pre-process that includes tokenization and turning tokens to lowercase, accepting only those that are present in the model's vocabulary. To finalize the computation, the model's method *cosine_similarities()* – existent for any of the mentioned models – is used between the input's and saying's sentence vector.

For the application of **BERT**, a pre-trained multilingual model of Bidirectional Encoder Representations from Transformers (BERT) was used, made available by Google and covering 104 languages, including Portuguese: BERT-Base, Multilingual Cased[5]. For the development of this method, a Python library named *bert-as-service*[6] was used, requiring to run a BERT server with the mentioned model, which is then connected to the client, i.e. to the system. This method has its own way to encode tokens, as it uses WordPiece tokenization, which may even divide tokens into sub-tokens, e.g. *walking* or *walker* become *walk@@ ing* and *walk@@ er*. In this specific hypothetical case, even if the model does not know how to deal with the word *walking*, it probably does know other words that have the *walk@@* in common, as it will appear more often. Because of this it only needed the computation of the sentence vector, which was made as in the previous methods, through the average of the vectors of its tokens. However, due to BERT specific encoding, the Cosine Similarity (CS) is calculated with the help of Python's *NumPy*[7] library. 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.

For most of the methods, both headlines and proverbs were first pre-processed with the NLPyPort package [Ferreira et al., 2019], a layer on top of 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 similarity methods, as seen above. Only BERT does not need this pre-process.

After the computation of any method, the similarity of each headline with each proverb is computed and the proverb with the higher similarity score is used. In this work, this is done for the eight tested methods, all relying in the computation of the CS between the vectors representing each sentence: these vectors were the result of the average of the vectors of the sentence's tokens. Following the computation of the similarity, the proverb with the best score is selected to represent the correspondent approach, as can be seen in Figure 4.2.

4.3 Evaluation

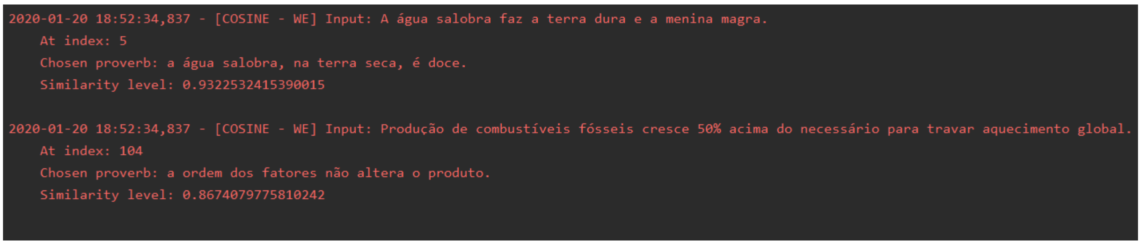
Evaluating the results of each method is a subjective task. Therefore, it was necessary to resort to human opinions, more precisely 24 volunteers who were asked to answer a

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

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

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

[7] https://numpy.org/



```
2020-01-20 18:52:34,837 - [COSINE - WE] Input: A água salobra faz a terra dura e a menina magra.
    At index: 5
    Chosen proverb: a água salobra, na terra seca, é doce.
    Similarity level: 0.9322532415390015

2020-01-20 18:52:34,837 - [COSINE - WE] Input: Produção de combustíveis fósseis cresce 50% acima do necessário para travar aquecimento global.
    At index: 104
    Chosen proverb: a ordem dos fatores não altera o produto.
    Similarity level: 0.8674079775810242
```

Figure 4.2: Example of a proverb selection, using a Glove model with 300 dimensions, and the calculation of the Cosine Similarity (CS) between the input and the chosen proverb WEs.

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 on Google Forms[8], for each of the ten assigned news headlines, the person answering had to classify the selected expressions with regard to two aspects: (i) relatedness, which classifies the semantic relation 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. Examples of questions of the survey are depicted in Figures 4.3. Only these two parameters were measured, as there was no need to validate the syntax (every expression was unaltered, only selected), and it did not seem pertinent to evaluate originality at this point, as the expressions were not generated by the system yet. For the scope of this evaluation, the judges did not have to justify the score they gave each expression, as it could influence their opinion.

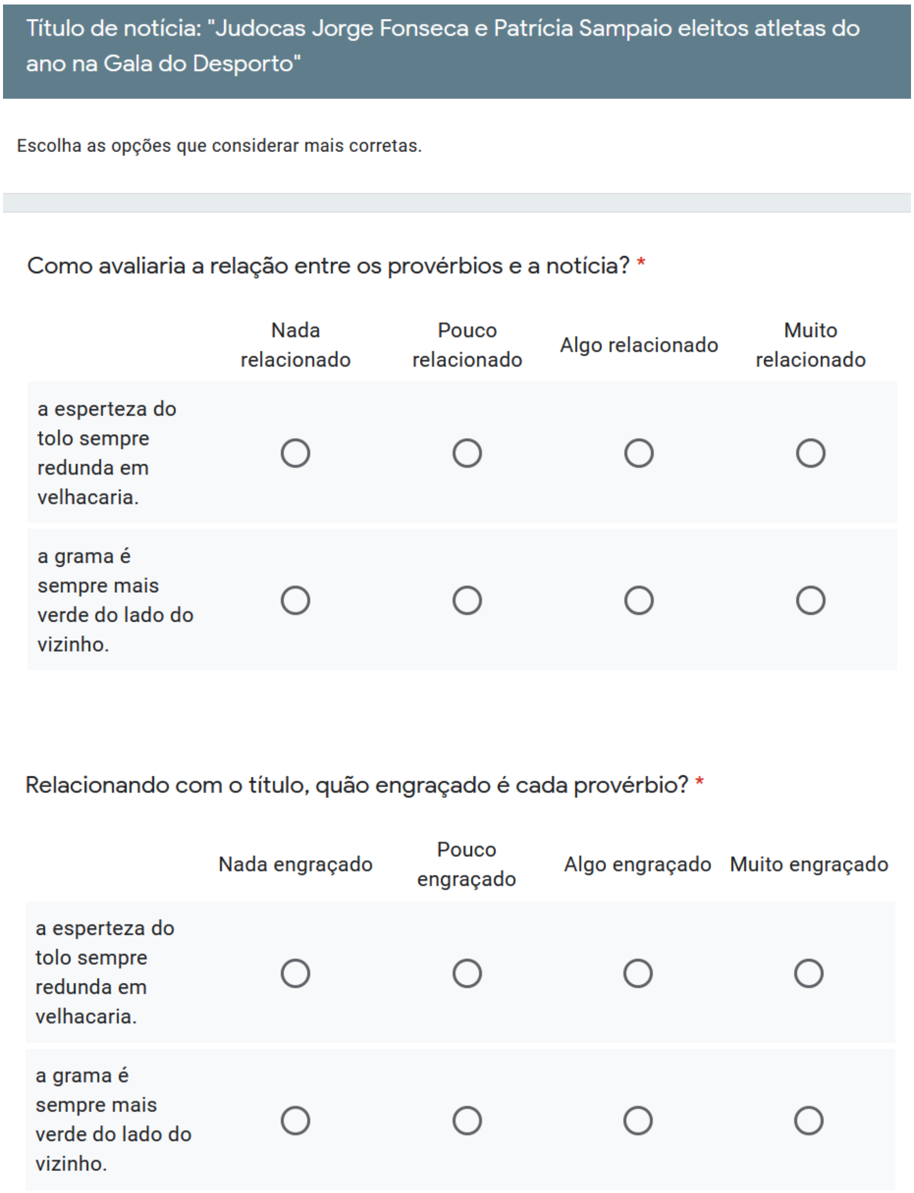
As an example, for the headline *"Emissões atmosféricas aumentaram em 2017"* ("Atmospheric emissions raised in 2017"), the volunteers were asked *"How would you rate the relation between the proverbs and the news title?"*. Below these questions, they would see the list of selected proverbs in a random order, with no repetitions, and rate each one according to a four point scale, which for relatedness was:

1. The expression is not related to the input;
2. The expression is remotely related to the input;
3. The expression is considerably related to the input;
4. The expression is extremely related to the input.

Funniness was evaluated under the question *"In relation to the headline, how funny is each proverb?"*. The presented four point scale had the following meaning:

1. The expression is not funny;
2. The expression is remotely funny;
3. The expression is considerably funny;
4. The expression is extremely funny.

[8] `https://www.google.com/forms/`



Título de notícia: "Judocas Jorge Fonseca e Patrícia Sampaio eleitos atletas do ano na Gala do Desporto"

Escolha as opções que considerar mais corretas.

Como avaliaria a relação entre os provérbios e a notícia? *

	Nada relacionado	Pouco relacionado	Algo relacionado	Muito relacionado
a esperteza do tolo sempre redunda em velhacaria.	○	○	○	○
a grama é sempre mais verde do lado do vizinho.	○	○	○	○

Relacionando com o título, quão engraçado é cada provérbio? *

	Nada engraçado	Pouco engraçado	Algo engraçado	Muito engraçado
a esperteza do tolo sempre redunda em velhacaria.	○	○	○	○
a grama é sempre mais verde do lado do vizinho.	○	○	○	○

Figure 4.3: Example of a survey question, with regard to the relatedness between headline and selected expression, and to the funniness of the expression.

In Table 4.1, the results of this evaluation are revealed, respectively for the relation between proverb and headline and its funniness. From this data, it was possible to compute certain metrics, like the mode (Mo), means (μ) and standard deviation (σ) and median (Md), to better understand the scope of these results.

Table 4.2 presents a global evaluation considering only the best proverb selections, which we considered to be those with at least a 3.5 average score on the relatedness ranking.

The Fleiss coefficient [Fleiss, 1971] is a metric that is similar to Cohen's Kappa coefficient [Landis and Koch, 1977], measuring inter-rater reliability between the responses and their own classification, also taking into consideration the possibility of judges agreeing by chance. However, Cohen's Kappa only works for two evaluators, while Fleiss' coefficient is able to measure more than two. As there were 6 different surveys, each with four evaluators, the metric was computed for each survey and each validated aspect: (i) Relatedness

Method	Relatedness			Funniness		
	Md	Mo	$\mu \pm \sigma$	Md	Mo	$\mu \pm \sigma$
Jaccard	2	1	2.4 ± 1.15	2	2	2.5 ± 1.06
Count	2	1	2.2 ± 1.06	2	2	2.3 ± 1.04
TFIDF	2	1	2.3 ± 1.09	2	2	2.3 ± 1.01
GloVe	2	1	2.2 ± 1.03	2	2	2.2 ± 1.02
GloVe + TFIDF	2	1	2.1 ± 1.05	2	3	2.2 ± 0.99
FastText	2	1	2.0 ± 1.03	2	1	2.0 ± 0.97
FastText + TFIDF	2	1	2.1 ± 1.03	2	1	2.2 ± 1.05
BERT	2	1	2.1 ± 1.04	2	1	2.2 ± 1.11

Table 4.1: Results regarding the relatedness (Rel) and funniness (Fun) of the expression with concern to the inputs.

Method	No. Selections	Mean	Median
Jaccard	7/60	2.6 ± 1.27	3
Count Vectorizer	3/60	3.7 ± 0.58	4
TF-IDF	6/60	2.3 ± 1.03	2
WE+GloVe	2/60	3.5 ± 2.12	3.5
WE+GloVe+TF-IDF	1/60	2	2
WE+FT	2/60	2.5 ± 0.7	2.5
WE+FT+TF-IDF	1/60	3	3
BERT	1/60	3	3

Table 4.2: Number of selections with more than 3.5 average score and the intersection of tokens between the proverb and the news title.

agreement was equal to 0.094; (ii) funniness agreement was equal to 0.054. To interpret these results, it possible to use Cohen's Table 4.3. The metric shows that there is just a slight agreement in terms of relatedness and funniness, proving the high subjectiveness of this kind of evaluations.

Cohen's Kappa	Interpretation – Strength of Agreement
< 0.00	Poor
0.00 – 0.20	Slight
0.21 – 0.40	Fair
0.41 – 0.60	Moderate
0.61 – 0.80	Substantial
0.81 – 1.00	Almost Perfect

Table 4.3: Landis and Koch interpretation of Cohen's Kappa values.

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 4.1, the majority of the results were not very satisfactory, as most of the proverbs had little relation to their headlines, as seen in the mode of each method. However, in regard to the funniness of the proverbs, the mode varied from method to method, producing some good results, which can mean that the idea of mixing proverbs with news headlines may give the headline a humorous touch.

From Table 4.1 and Table 4.2, it is possible to declare that theoretically simpler methods achieved the best scores both in the quality of the relation between the selected proverb and the given headline, particularly the simplest one, the Jaccard similarity.

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 relation between them. The same can be

News Headline + Proverb	**Method**	μ Rel	μ Fun
"Malásia devolve 150 contentores ilegais de lixo a países subdesenvolvidos" ("Malaysia returns 150 illegal trash containers to underdeveloped countries") + *"Quem faz de si lixo, pisam-no as galinhas"* ("Whoever makes trash out of you, the chickens will stomp them")	TF-IDF	4	3.5
"Tempestade 'Glória' fez 12 mortos em Espanha. Governo culpa alterações climáticas" ("'Gloria' storm made 12 casualities in Spain. Government blames climate change") + *"A culpa morre solteira."* ("Guilt dies single")	TF-IDF	4	3.25
“Ainda não é demasiado tarde para salvarmos os oceanos” ("It is not too late to save the oceans") + *"Não deixe para amanhã o que você pode fazer hoje."* ("Do not leave for tomorrow what you can do today")	BERT	4	2.5
"Veredicto abre a porta a protecção para 'refugiados climáticos'" ("Veredict opens door for protection to 'climate refugees'") + *"Para trás mija a burra."* ("The donkey urinates backwards")	Jaccard	2.5	4
"Judoca Jorge Fonseca galardoado com o prémio Ética no Desporto de 2019" ("Judoka Jorge Fonseca is awarded with the prize 'Sport Ethics 2019'") + *"Não contes com o ovo no cu da galinha."* ("Do not count on the egg being in the chicken's butt.")	Jaccard	1.5	4

Table 4.4: Some expression selections whose relatedness between headline and saying, or funniness, was rated 4.

said for the second and third methods, the Count Vectorizer and the TF-IDF Vectorizer, which also achieved the next best two scores.

A curious fact is that the more recent methods, 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 different from their title's meaning, their relation is, perhaps, not as pinpointed or clear as in the simpler methods. However, despite having lower scores, BERT was able to make a selection which scored an average of 4 points, as seen in the third headline in Table 4.4, where the chosen proverb's meaning and urgency clearly applies and is related to the headline.

Another one of the best examples of a successful proverb selection regarding the relation between title and proverb, is the first headline in Table 4.4, which scored 4 for all its 4 testers. Using the TF-IDF Vectorizer method, 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 headline of Table 4.4, 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]. However, it is also noticeable that even though the usage of taboo words might increase funniness, the relatedness scores were not high, thus raising the division betweem these two aspects and their compatibility.

On the other hand, two of the selections with the worst rate in terms of both relatedness and funniness are depicted in Table 4.5. Both of these examples were obtained with WEs, even though the first used the GloVe model, while the latter used the FastText model. The first example presented tries to make use of the existence of a number and other kind of content 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.

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

Table 4.5: Some expression selections whose relatedness and funniness was rated 1.

Table 4.2 is an indication of the methods with higher global success in terms of 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 the lead is the simplest algorithm, the Jaccard similarity, followed closely by TF-IDF Vectorizer and Count Vectorizer. In regard to the amount of words shared between these chosen proverbs and the news title they are related to, the median of this statistic is higher for the Count Vectorizer and Jaccard. Furthermore, the more complex and recent approaches did not have as many successful selections. However, the chosen selections of these approaches coincided with proverbs with at least two common words with the headline. Therefore, it may be possible to argue that people are quicker to relate two sentences that share the most words, in regards to their full semantic value chosen by the approaches that are more up to date, according to this data.

4.4 Discussion

In this Chapter, the methods for selection of text based on context were detailed, from a recap on the theory behind them in Section 4.1 to their development implementation in this system in Section 4.2. An evaluation of these methods was also described in Section 4.3, which was followed by a discussion of results in Section 4.4.

As for the results of the evaluation, the relatedness aspect could have had better scores, but it is dependent on the topics of the input's headline, which can be about virtually anything. However, these results motivate the second goal of this thesis, to generate adapted expressions to any headline, creating a novel and original, and more related to the input.

The study described in this chapter was submitted and accepted as a full paper in International Conference on Computational Creativity (ICCC), and was added *a posteriori* to the Texto Em Contexto (TECo) system architecture, which will be discussed in Chapter 6.

Chapter 5

Adapting short-Text based on Context

The previous chapter focused on the first half of the work, a short-text selector based on context. This chapter presents the set of three proposed methods for adapting Portuguese expressions in Section 5.1, aiming to achieve higher relatedness to the input: (i) Substitution; (ii) Analogy; (iii) Vector Difference. All the adaptation methods identify relevant words of the input and get representations of their meaning from Word Embeddings (WEs). Relevant words in a known expression are then replaced by words in the input or related to them, thus producing an output theoretically more related to the input than the original expression.

A first attempt to validate results is described, which was also useful to find and fix some minor issues in the implementation of the methods. The necessary setup is described in Section 5.2, both in terms of collecting data and the methodology utilized for the implementation. Afterwards, it dives into the description of the preliminary validation in Section 5.3, presenting and discussing its results. Finally, Section 5.4 recapitulates on this chapter's content and draws some conclusions based on the preliminary validation's results.

5.1 Generation Methods

The purpose of this Section is to detail the three proposed automatic methods for the adaptation sub-system: (i) Substitution; (ii) Analogy; (iii) Vector Difference. These methods share some common ground in between them, namely:

- All methods exploit a pre-trained model of static Word Embeddings (WEs), where words are represented by dense numeric vectors, as previously introduced in Section 2.4. This is in line with related works that exploit vector representations of words in the transformation of text, e.g. the work developed by Bay et al. [2017].
- They compute the most relevant words in a text, which can be done with methods like Term Frequency - Inverse Document Frequency (TF-IDF), or simply by considering the open-class words (nouns, verbs, adjectives) that are used in a large corpus but have the lowest frequency.
- They go through all the sayings in a constrained set and try to make adaptations focused on the most relevant words of both the sayings and the input texts.

The main difference between the generation methods are the adopted strategies for selecting the word(s) to replace. To avoid syntactic inconsistencies, for any method, replacement candidates must match the morphology of the replaced word, including part of speech (PoS), gender and number, e.g. if a word named a is a noun, singular and feminine, the substitute will have to match these characteristics, obtained from a morphology lexicon and / or by the application of a PoS tagger.

Nonetheless, if the morphology of the words within the comparison does not match, the lexicon can also be used to inflect the candidate to the target form, by reducing it to its base form – *lemma*, see Section 2.3 – and search in the lexicon for any words that share the same lemma of the candidate and the morphology of the word to be replaced, e.g. the verb *"arriscaram"* can be inflected to the lemma *"arriscar"*, shared by many other variations of the verb. If this operation is not successful, the candidate is not considered, possibly leaving the saying itself out of consideration, depending on the chosen method. In any case, the set of possible replacements can be augmented by considering not only the relevant words in the input text, but also the most semantically-similar words.

In addition to morphology constraints, it was also tested to restrict the comparison only to words that share the same number of syllables, in an attempt to keep the structure of the original expression as much a possible. However, this ended up as a big constraint, hampering the system's ability to produce outputs, and the ones it did produce did not seem to manifest an increase in quality.

Finally, running through all available sayings should result in several new texts. Even if, due to the morphology constraints, some sayings end up not being used, others will. Also, if similar words are considered for the same input, the same method may produce several variations of the same text. As each method generates several new expressions (possibly even several adaptations for each saying), the sub-system relies on the selector from Chapter 4 to make the final decision between the several possibilities of expressions they build.

5.1.1 Substitution Method

The first method, named Substitution, replaces the most relevant word in the saying, a, by the substitute, b, which ranks best between a constructed list of several possibilities. The reasoning behind this choice was that, by using a relevant word w of the input text or a similar word to w, the meaning of the saying becomes more semantically-related to the given context. This approach is similar to the lexical replacement developed by Gatti et al. [2015], seen in Chapter 3.

As such, by considering the most relevant keywords from the input, the second step was to make a list of all possible candidates for replacement. In pursuance of creating such list, for each keyword from the input, a partial list is computed, composed by the n most similar words with regard to the each keyword given as input, based on the CS. Therefore, if the headline has m keywords, m partial lists are created. To make the transition from the partial list to the full list of candidates, each candidate was ranked based on its position within its partial list – sorted from the largest CS value to the lowest – and the number of occurrences in different partial lists, i.e. the same word may be a possible substitute for more than one keyword from the input and there was a need to avoid duplicates while also strengthening these words rank, since their reappearance was taken to be a signal of increased relatedness to the input's context.

The full list of candidates is then reduced to only those whose morphology – PoS, gender and number – match the characteristics from a. This operation is computed using the morphology lexicon and PoS tagger. From the final list of candidates, the system proceeds to replace a with each possible b, generating several new expressions that are appended to the list of generated expressions that is finally sent to the pre-defined selector.

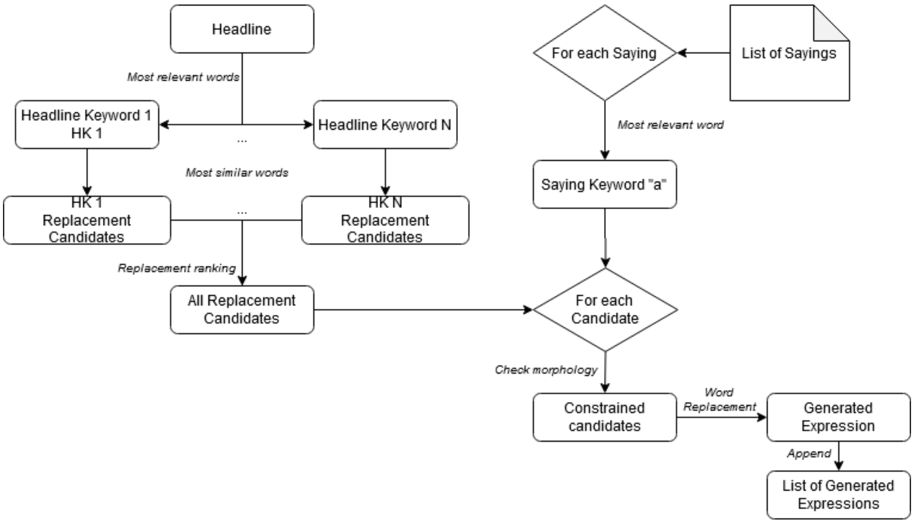


Figure 5.1: Substitution generation method flow.

The flow of computations present in this method is represented in Figure 5.1. As an example, by considering as input the headline *"Bancos preparam-se para dar menos crédito às famílias"* (Banks are preparing themselves to give less credit to families), this method generated the expression *"Bancos, bancos, negócios à parte."* (Banks, banks, business apart), which was created from the Portuguese proverb *"Amigos, amigos, negócios à parte."* (Friends, friends, business apart) by the substitution of the noun *"amigos"* (friends) with the noun *"bancos"* (banks). The process of generating this example is depicted in Figure 5.2.

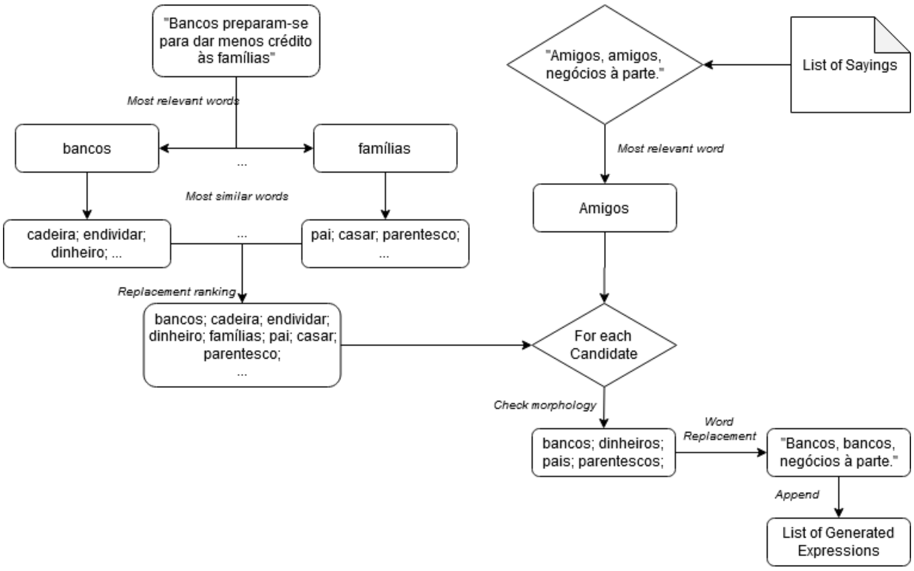


Figure 5.2: Example of the Substitution generation method flow.

5.1.2 Analogy Method

The second method, Analogy, relies on a common operation for computing analogies in WEs, known as 3CosAdd [Bay et al., 2017], i.e., $b - a + a^* = b^*$, which can be phrased as b^* *is to* b *as* a^* *is to* a, already discussed in Section 2.4, where the example $king - man + woman = queen$ was given to exemplify this notion.

The strategy is to use the two most relevant words in the saying as a and a^*, and the most relevant word in the input as b, obtained by verifying their frequency in the newspaper corpus. Then:

1. From the previous three words a, a^*, and b, compute a new word b^*, considering the CS. The chosen word has to match the morphology of a^*, which is guaranteed by the utilization of the already mentioned morphology lexicon.

2. In the original saying, replace a and a^* by b and b^*. Given that both pairs of words are analogously-related, our intuition is that the result will still make sense and be more related to the input text. It is important to note that the method also allows only one replacement, i.e. in case b^* does not match the morphology of its match-up word in the saying (e.g. if b^* is a noun and is candidate to replace a, which is a verb, while both b and a^* are both nouns with a successful match-up).

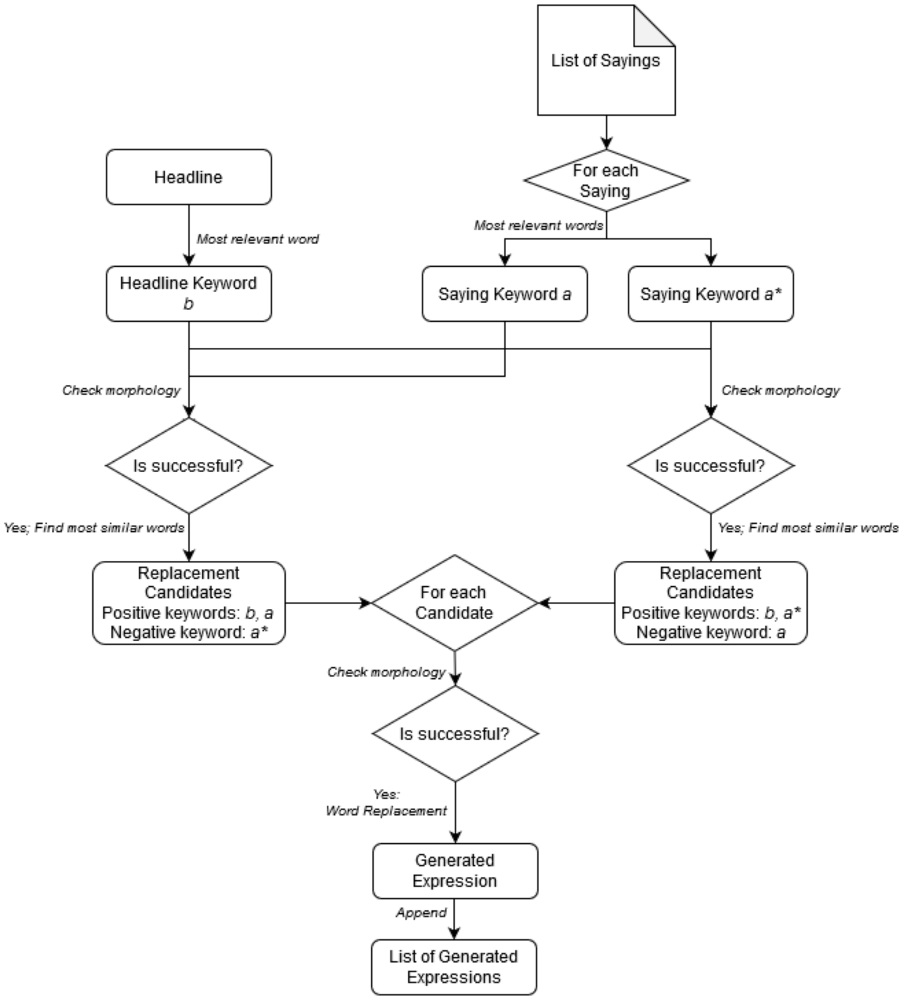


Figure 5.3: Analogy generation method flow.

The flow of this method, depicted in Figure 5.3, starts by choosing the most relevant word in the input, b, and the two most pertinent words, a and a^*, in the saying. Afterwards, it compares the morphological characteristics – PoS, gender and number – of b with the ones of a and a^* and, if there are no matches possible, the saying is not used. However, if at least one match is verified, more substitutes are computed, considering b as positive input and its matching word as negative. Considering that the matching word is a, for each word in this set of replacement candidates, there is a verification between its morphology and the one of a^*. If there is a match, a new expression will be generated by replacing a with b and a^* with the candidate. Nevertheless, if there is no substitute to match the morphology of a^*, then the only replacement to be done between a and b, keeping a^*, which is a new expression by itself, e.g. if a^* is a feminine noun in single form and there is no b^* that matches this morphology, then a^* is kept.

For the input headline *"EUA estão a apontar para o pior número de desemprego da sua história"* (USA are pointing to the worst unemployment numbers of their history), this method generated *"Não comeces para amanhã o que podes apontar hoje"* (Do not start tomorrow what you can point today), from the saying *"Não deixes para amanhã o que podes fazer hoje"* (Do not leave for tomorrow what can be done today). In this example, a and a^* are represented by the verbs *"deixes"* (leave) and *"fazer"* (do), while b^* and b are represented by the verbs *"comeces"* (start) and *"apontar"* (point), the latter being directly descendant from the input headline. The flow, with regard to this example, is presented in Figure 5.4.

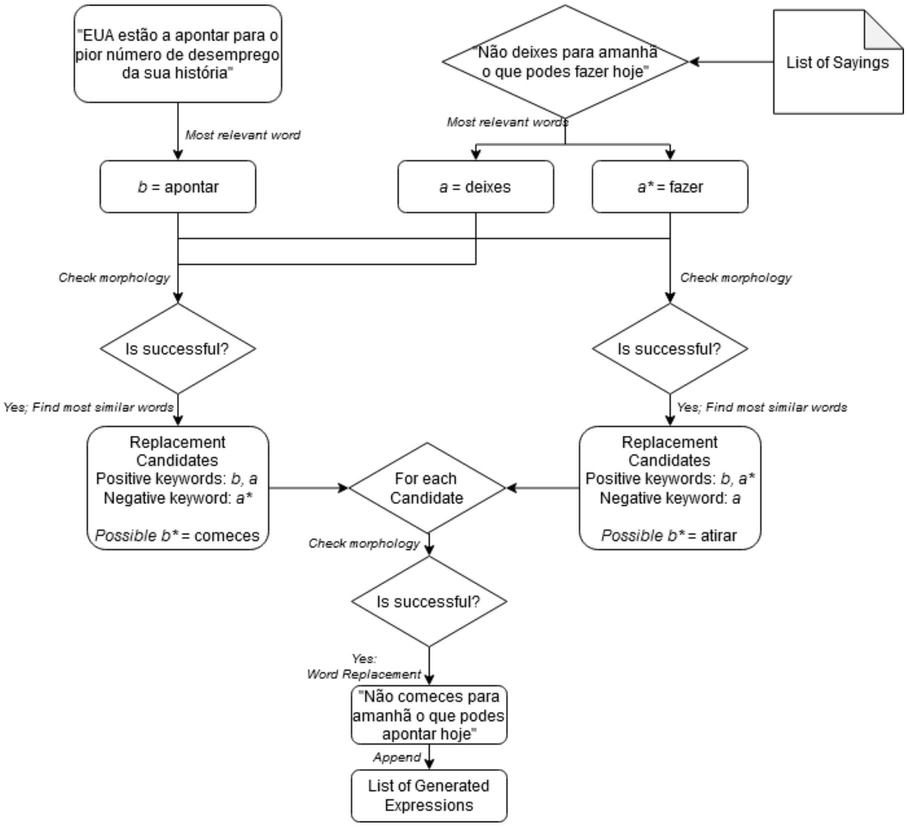


Figure 5.4: Example of an Analogy generation method flow.

5.1.3 Vector Difference Method

The third method, Vector Difference, selects the two most relevant words in the input text, b and b^*, and considers the relation between b and b^*. The flow of this method, represented in Figure 5.5, follows these guidelines:

1. Compute the vector of the relation between b and b^*, through the subtraction of their own vector representations (WEs), depicted as $b - b^*$.
2. For each saying, identify pairs of open-class words in the saying, a and a^*, that match the morphology of b and b^*, i.e. a has to match either b or b^*, with a^* matching the other remaining one, regardless of the order of appearance.
3. If there is more than one pair that matches the morphology of b and b^*, choose the pair whose relation, $a - a^*$, maximizes the CS with $b - b^*$.
4. Replace a and a^* respectively by b and b^*, maintaining the morphology characteristics in each of the replaced words, i.e. if a matches b and a^* matches b^*, the replacement occurs in that order.

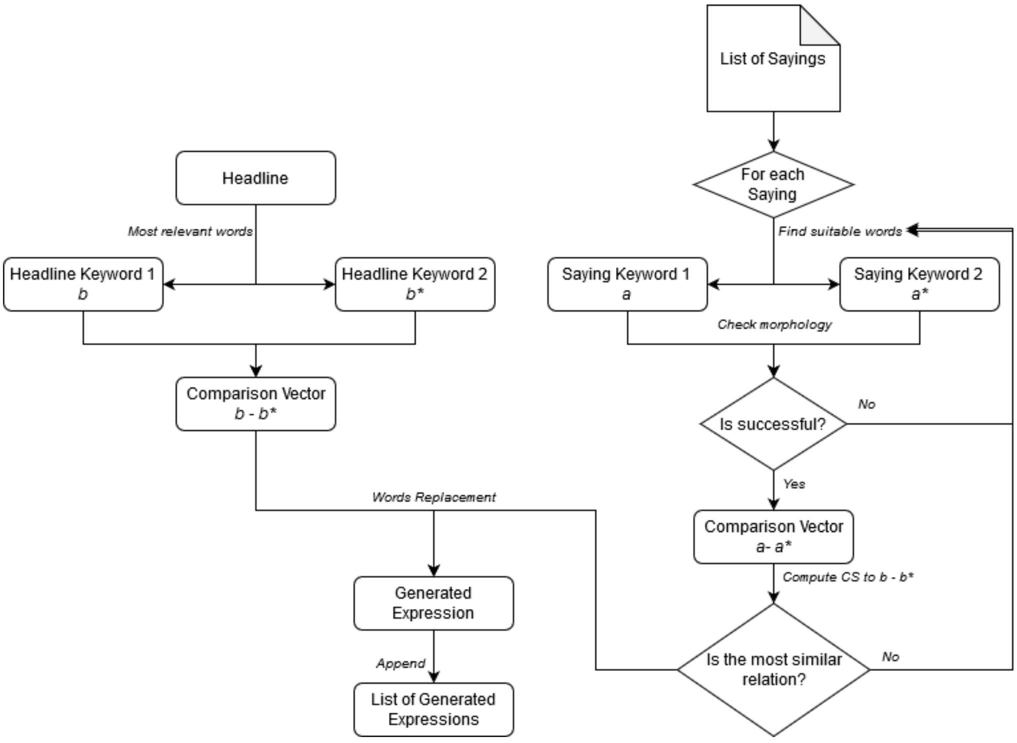


Figure 5.5: Vector Difference generation method flow.

The reasoning underlying the development of this method is that replacing two of the saying's words with two words directly from the input will increase the relatedness between both expressions and possibly create some unexpected contextual twist. Moreover, the input words will be included in such a way that the relation between them is roughly preserved.

For the input headline *"Finge ter Covid-19 no Facebook e acaba detido"* (Pretends to have Covid-19 on Facebook and ends up arrested), this method generated the expression *"Quem com ferro finge, com ferro será detido"* (Those that pretend with iron, with iron shall be arrested), which descends from the saying *"Quem com ferro fere, com ferro será* *ferido"* (Those who hurt with iron, with iron shall be hurt). This method replaced each usage of the verb *"ferir"* (hurt) – *"fere"* and *"ferido"* – by the verbs in the headline, *"finge"* (pretend) and *"detido"* (arrested). The flow of this example, is presented in Figure 5.6.

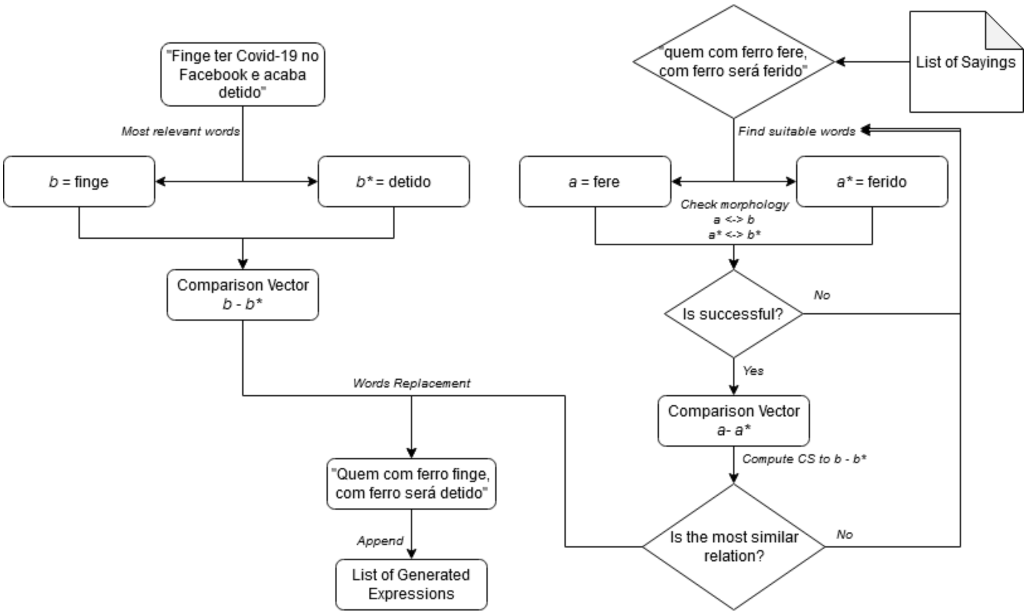

"Finge ter Covid-19 no Facebook e acaba detido"
Most relevant words
b = finge
b* = detido
Comparison Vector b - b*
Words Replacement
"Quem com ferro finge, com ferro será detido"
Append
List of Generated Expressions
"quem com ferro fere, com ferro será ferido"
List of Sayings
Find suitable words
a = fere
a* = ferido
Check morphology a <-> b a* <-> b*
Is successful?
No
Yes
Comparison Vector a- a*
Compute CS to b - b*
Is the most similar relation?
No

Figure 5.6: Example of a Vector Difference generation method flow.

5.2 Experimental Setup

Before presenting the full display of results and their respective evaluations, a first evaluation was necessary to validate and identify possible problems, in an attempt to correct them before doing the final evaluation. Moreover, this preliminary evaluation was also useful on the submitted short-paper to the International Conference on Computational Creativity (ICCC), and is described on Section 5.3. This allowed a final verification of the system, before presenting the results to a larger group of human evaluators.

Rewinding to previous chapters, it is important to note that both selection methods get the most similar sayings to the input context. The difference is that TF-IDF represents each text as a weighted vector, based on all the sayings, while BERT encodes each text as a 768-sized vector according to a pre-trained model.

As described in Section 4.2, this system was developed in Python 3.6, using its many available libraries, both for text related operations and for statistical purposes.

There was an obvious need for a good collection of data with defined context, accomplished by gathering news headlines from different sources. In the beginning, they were collected through the News API[1], as said in Section 4.2. In later stages of development, the creation of a bot developed for Twitter, named *@TextoEmContexto*, allowed a wider search in this social network, as described in Section 6.3.

Alongside the news, it is important to have a large enough corpus of expressions: the used corpus included 1,600 Portuguese proverbs from project Natura[2] and over 2,500 movie

[1] `https://newsapi.org/`

[2] `https://natura.di.uminho.pt`

titles in Portuguese, from IMDB[3]. Most should be well-known, with proverbs being part of the quotidian of most Portuguese people, where they are used to emphasize certain situations, usually implying some kind of humour. Moreover, these sayings are not usually to be taken literally, as they employ several stylistic variations and figurative language, and their underlying meaning may not be straightforwardly understood by a computer. With regard to the movie titles, they were constrained from a list of over two million movie titles, considering only the 75% most recent movies, that were in Portuguese and had four or more words, all of which needed to be present in the morphology lexicon. This was a security measure to avoid unknown words for the Portuguese language, e.g. *Indiana Jones*, as the system would not be able to process them.

For the purpose of selecting the most relevant words, the system relied on the word's frequency in the newspaper corpus CETEMPúblico [Rocha and Santos, 2000]. Concerning the morphology comparison between words, their characteristics were obtained from a morphology lexicon, LABEL-Lex [Ranchhod et al., 1999], and / or complemented by the application of a PoS tagger, for which NLPyPort [Ferreira et al., 2019] was used. The choice of using both approaches was due to the possible ambiguity present in some words that may have a distinct PoS in different contexts. This ambiguous situation can be exemplified by the word *"risco"* which, depending on the context, may be referring to the noun *risk*, synonym of a situation that might involve danger, or the verb *scratch*, as in scratching a paper with a pencil. In such cases, the morphology lexicon gives more than a possible PoS for the word and, as such, needs the help of the PoS tagger for disambiguation. On the other hand, due to performance issues regarding time and quality of results for the simpler cases (most of the words are not ambiguous), the lexicon was chosen to make the primary decisions with regard to each word. Only if this operation is not successful or does not produce any certified results, does TECo rely on the PoS tagger.

Concerning the retrieval of the most semantically similar words, they were represented with WEs, using a GloVe model with 300 dimensions, which has been proven to give better results in terms of the computation of semantic analogies than FastText, both for Portuguese [Hartmann et al., 2017, Sousa et al., 2020] and for English [Drozd et al., 2016], particularly in the application of the 3CosAdd method $king - man + woman = queen$ [Mikolov et al., 2013b]. This representation permits the computation of the *most_similar()* method that belongs to Python's Gensim library[4]. This method returns a list of the n most similar words to the input, regardless of morphology, through the usage of the Cosine Similarity (CS) metric. It receives as input a list of positive words for which it aims to measure similarity, and a list of negative words to influence the measurement in a way that the returned words are distinct from the negative input. For example, let's consider we input the word *coffee* in this method, it would return a list of words that may share some attributes with the input, like *tea* or *juice*, which are also beverages and thus similar to *coffee.* With regard to BERT, the used pre-trained model covered 104 languages[5].

5.3 Preliminary Validation

This first evaluation aimed to validate the methods, and identify and fix minor issues that might appear. It was also used to write the previously mentioned short-paper, submitted in International Conference on Computational Creativity (ICCC). It used the corpus of proverbs and movie titles described in Section 5.2, and 30 news headlines retrieved from

[3] `https://www.imdb.com/interfaces/`

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

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

tweets from Twitter accounts of several Portuguese newspapers. For each headline, a new expression was produced by each one of the three adaptation methods and the results were manually assessed, independently, by two human judges. Considering the amount of expressions from which to select and adapt, an initial selection of 90 was made, where each of the selection methods TF-IDF and BERT selected 30 each, with the last 30 being randomly selected from the entire corpus, to increase diversity.

For each headline, judges were presented with the headline followed by the list of texts produced by each of 8 combination of methods: (i) Combining each of the 3 generation methods with each of the 2 selection methods; (ii) Outputs that did not suffer any adaptation and were selected by TF-IDF or BERT. For each expression, the judges were asked to use a 3-point Likert scale for ranking, considering three different aspects that are key for the quality and suitability of the resulting text: (i) syntax; (ii) relatedness; (iii) funniness. Judges did not have to justify their answers in any of the aspects, as this would have a direct influence on the evaluation.

The **syntax** grading aspect was meant to classify the amount of grammatical and/or structural issues that the expressions had. Once the adaptation methods replace words in the text, automatically, if care is not taken (e.g., going through morphology checks), problems can easily arise and lead to text with odd grammatical structure, also hard to interpret. Having this in mind, the following scores could be given to the syntax:

1. The expression has several grammatical and/or structural issues, and may be difficult to interpret;
2. The expression has minor issues regarding grammar and structure, but is still understandable;
3. The expression does not have any grammatical or structural issues.

As in the previous evaluation, **relatedness** assesses the relation between the semantics of the resulting text with the semantics of the input. A very low relatedness suggests a random semantics in the generated text, meaning that the goal of producing text that is indeed related to the context simply falls. Therefore, it is a key aspect. Moreover, we tackled the adaptation of text as a way to increase precisely relatedness, so we were expecting that it was higher for adapted texts than when existing expressions are just selected. Relatedness was classified according to the following scale:

1. There is minimal or no relation at all between the generated expression and the input;
2. The expression is somewhat related to the input, as it shares one or two words and its context is somewhat relatable to the one of the input;
3. The expression is clearly related to the input's context and may be of use as a candidate to replace the input itself or as a comment to the input.

Finally, the last aspect to be validated was the **funniness** of the generated expression. As said before, this is an extremely subjective evaluation and depends a lot on the opinion of the judge, as it may be drastically different from one person to another. It is possible to find an expression funny regardless of the input, while it is also possible to find it funny due to its relation to the input, and both are acceptable. The scoring possibilities for the funniness aspect were the following:

1. The expression is not funny and will most likely not make anyone laugh;

2. The expression is somewhat funny or could be funny, depending on the reader's subjective view;
3. The expression is very funny, and has a great potential to make other people laugh.

Table 5.1 shows several figures of this preliminary evaluation, namely the mode (Mo), means (μ) and standard deviation (σ), and median (Md) of the scores for texts produced by each adaptation method, regardless of the judge and selection methods, plus the results of the two selection methods when applied directly to the full list of sayings.

Method	**Syntax**			**Relatedness**			**Funniness**		
	Mo	$\mu \pm \sigma$	**Md**	**Mo**	$\mu \pm \sigma$	**Md**	**Mo**	$\mu \pm \sigma$	**Md**
Substitution	3	2.94 ± 0.24	3	3	2.06 ± 0.88	2	2	1.71 ± 0.68	2
Analogy	3	2.91 ± 0.28	3	3	2.31 ± 0.76	2.5	1	1.64 ± 0.66	2
Vector Diff	3	2.87 ± 0.33	3	3	2.51 ± 0.72	3	2	1.81 ± 0.64	2
TF-IDF	3	3.00 ± 0.00	3	3	2.35 ± 0.83	3	1	1.62 ± 0.66	2
BERT	3	3.00 ± 0.00	3	1	1.70 ± 0.90	1	1	1.47 ± 0.64	1

Table 5.1: Preliminary validation figures.

Judge agreement was measured with **Cohen's Kappa coefficient**, used to measure the inter-rater reliability between the responses and their own classification, also taking into consideration the possibility of judges agreeing by chance. It resulted in 0.57 for syntax, 0.35 for relatedness and 0.17 for funniness, which can be interpreted with aid from the standard guidelines to interpret the coefficient's results, presented in Table 4.3, which were defined by Landis and Koch [1977]. It is then possible to name the agreement between raters as *moderate* for the syntax and *fair* for the relatedness and funniness.

Syntax is clearly the less subjective aspect, because it is more or less straightforward to find text with inconsistent grammar (see Table 5.2). Thus, agreement was higher. On the other hand, the other two aspects, especially funniness, are highly subjective, also due to the structure and figurative language of the Portuguese proverbs and vagueness of some movie titles. According to the scores, the syntax is not severely affected by the adaptations, meaning that the produced text is generally grammatically correct. Few exceptions occur in the adaptation of verbs. Specifically, in Portuguese, the same verb has often different forms for different tenses, genders and numbers, but the same form may also work for different tenses. Thus, incorrectly identifying the tense in the original saying may result in using an incorrect form in the adapted text, like the ones presented in Table 5.2. In the first example, the problem was the ambiguity of the word *"teste"* (test), that can be a noun or a verb in very similar contexts, thus being difficult to identify its PoS, which in this case was considered as a noun when it should be a verb, making the replacement by the noun *"ano"* (year) a mistake. On the second example, the problem is the verb form, as *"é"* (is) is in the third person of the Present, while *"mantém"* (keeps) is also in the third person of the Present, but requires the use of the pronoun *"se"* (itself) to be morphologically correct.

Original Expression	**Generated Expression**
"Teste o seu cérebro" (Test your brain)	*"Ano o seu cérebro"* (Year your brain)
"A grama é sempre mais verde do lado do vizinho" (The grass is always greener on the neighbor's side)	*"A grama mantém sempre mais verde do lado do vizinho"* (The grass always keeps greener on the neighbor's side)

Table 5.2: Two examples of syntatic problems.

Regarding relatedness, results shown in Table 5.1 are above average, i.e. means higher than 2 and mode equals 3. This means that a relation between input and output was noticeable, even if the results could be higher, and the generation is far from being random. Despite a slight trend of Vector Difference getting higher scores and Substitution lower, especially looking at the median, the standard deviation also suggests that this difference is not significant enough. Though, differences would make sense because, while Substitution makes a single replacement, the other two methods replace two words that, nevertheless, try to keep the original relation, with the Vector Difference generally using two words from the original context.

On funniness, results were not as good as for the other aspects. This may be due to the aforementioned subjectivity of humour, which may hamper the judge's decision to give the maximum funniness to a text, because actually making other people laugh depends on many variables. Moreover, the capability of producing content with humoristic value is highly dependent on the context of the input, e.g. it is harder to generate content regarding sad news headlines, like *"Kobe Bryant, o fim trágico de uma estrela"* ("Kobe Bryant, the tragic end of a star").

5.4 Discussion

This chapter presented a detailed explanation of the three proposed generation methods: (i) Substitution; (ii) Analogy; (iii) Vector Difference. All these methods exploited the use of WEs and have similar structures, differing their mode of choosing and replacing words in the expressions. Moreover, Section 5.2 demonstrates how the work was developed, from the collection of data to the implementation.

Section 5.3 presented a preliminary validation of results, aimed to not only validate the methods, but also identify and fix minor issues that might appear. In addition, it was also used to write the previously mentioned short-paper, submitted in International Conference on Computational Creativity (ICCC). This validation focused on three aspects: (i) syntax; (ii) relatedness; (iii) funniness. Considering that syntax is the less subjective aspect, it was also the one that had higher scores and higher agreement between the evaluators. In terms of relatedness, there was an increase of scores when the adaptation methods were used as a complement to the selection methods, proving their usefulness as tools to increase the relation between the input and the generated expression. On the other hand, funniness had lower results, which go accordingly to what was expected, due to this aspect's subjectivity and to its dependency on the input's context, as not all contexts require/allow the generation of humoristic content.

Chapter 6

TECo: Text in context

To amplify the range of a given story, either real or made-up, authors commonly reuse expressions or sayings known by a general audience as a title or subtitle, sometimes also achieving a humorous effect. If the expression is related enough, it can be used directly, but it may also suffer minor adaptations, to become more related to the context and still resemble the original saying.

In this chapter, the fully developed work of this thesis is presented, namely a set of methods for adapting Portuguese expressions in order to achieve higher relatedness to the input. All the adaptation methods identify relevant words of the input and get representations of their meaning from Word Embeddings (WEs). Relevant words in a known expression are then replaced by words in the input or related to them, thus producing an output theoretically more related to the input than the original expression. The selection of those words depends on the method. The selection methods presented earlier also play an important role on the selection of the adaptation to return. Moreover, results of two of the latter methods are included in the evaluation. The developed methods are further integrated in a system for amplifying the range of news stories with creative text, named Texto Em Contexto (TECo) (Text within Context).

This chapter starts by putting it all together, with the system's architecture in Section 6.1. Since the experimental setup necessary for the final evaluation of TECo's results was the same presented in Section 5.2, this chapter dives directly into Section 6.2. This section presents the systematic validation process that was presented to several human judges in order to evaluate the system's results in three aspects: (i) relatedness; (ii) originality; (iii) funniness. Furthermore, it discusses the results of this final evaluation, drawing some important conclusions regarding the used approaches in this thesis.

Section 6.3 presents the Twitter bot *@TextoEmContexto*[1], along with its structure and implementation. This bot was used not only to retrieve data in the form of tweets, but also to publish some of TECo's generated expressions in this account.

Finally, Section 6.4 recapitulates on what was said in this chapter, discussing the main findings of the evaluation and some possible conclusions that can be drawn from it.

[1] `https://twitter.com/TextoEmContexto`

6.1 System's Architecture

As mentioned in Chapter 1, TECo is based on two sub-systems, one for selection and one for adapting/generating new expressions, with both operations relying heavily in the input's context. Considering the selection sub-system, the chosen algorithms were TF-IDF Vectorizer, due to the evaluation presented in Section 4.3, and Bidirectional Encoder Representations from Transformers (BERT), due to its considerably different approach, while also being a state-of-the-art method, thus increasing the interest in evaluating its performance.

Figure 6.1 presents a high level depiction of the work flow of the full TECo's process, where the selection and adaptation methods are integrated. It does not present details on any of the two sub-systems, as they are explained separately. The process starts by making an initial selection of sayings from the full dataset – always considering the input's context – for performance purposes: (i) Select the same number of sayings with both TF-IDF Vectorizer and BERT; (ii) A set of random sayings. Both of these selections avoid repeated sayings, and their simultaneous use was a measure that aimed to increase diversity in terms of the texts to be adapted.

From this point on, there is a set of expressions that will be subjected to adaptation. For each expression, generation methods, presented in Section 5.1, produce a set of several distinct new expressions. To decide the final output, this set of candidates is submitted to another selection, this time with a predefined method from the two already mentioned, TF-IDF Vectorizer and BERT. Therefore, there is always a connection between the chosen generation method and the selection method, the pair considered to be a combination of methods henceforth.

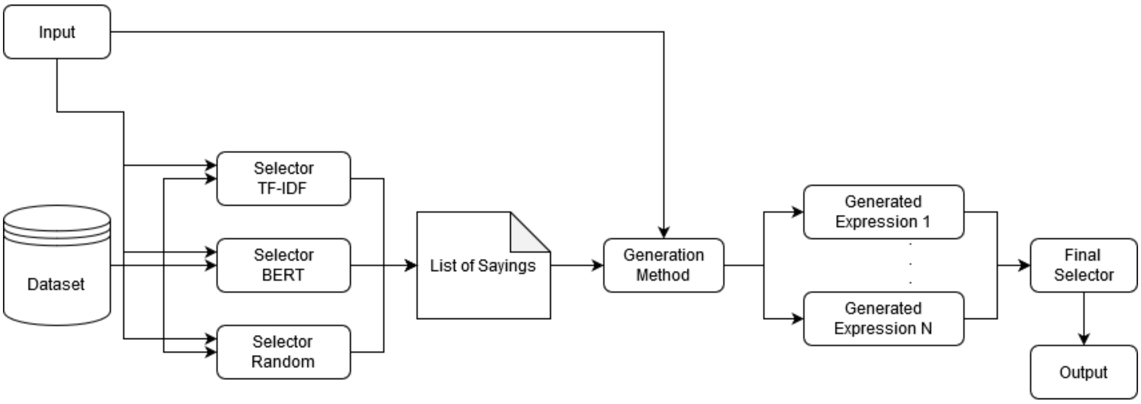
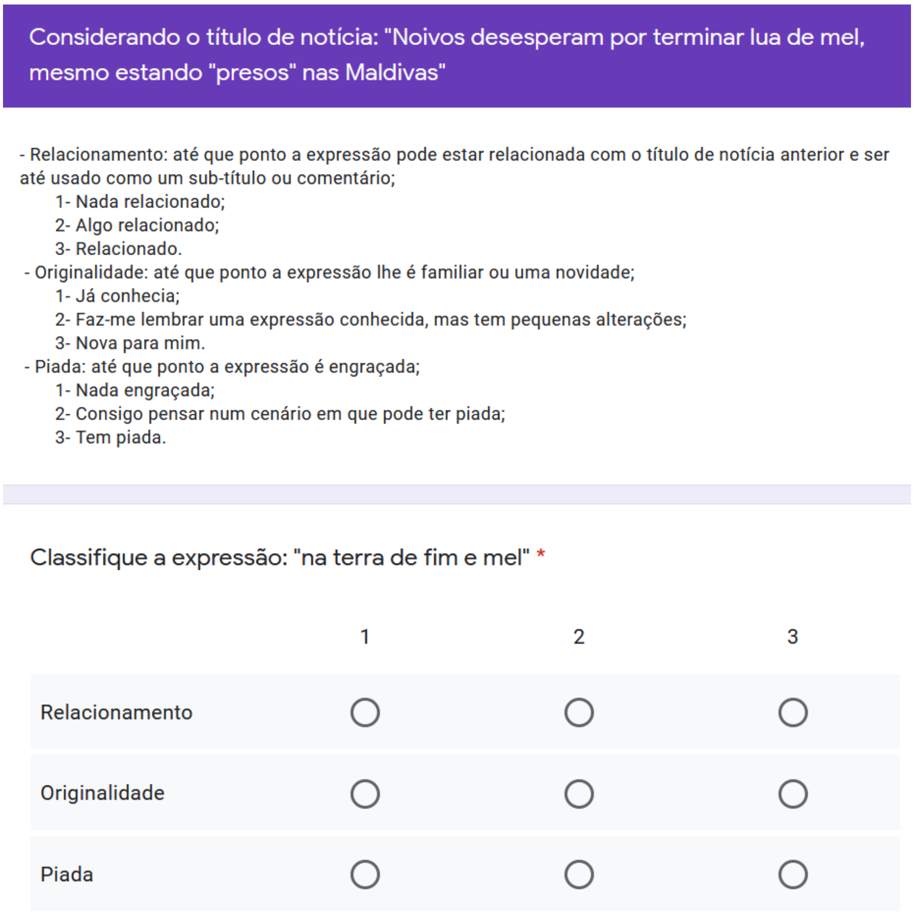


Figure 6.1: TECo's architecture.

6.2 Evaluation

For the purpose of validating TECo, a final evaluation was made to draw conclusions about the implemented methods. In terms of setting up the evaluation, the same data collected for the preliminary validation described in Section 5.1 was used. As such, 100 news headlines, retrieved from Twitter, were submitted as input for the system. With regard to the initial selection of expressions to be adapted, as was previously mentioned in Section 5.1, the methodology was repeated: from the total corpus of expressions, 30 were selected by TF-IDF, 30 by BERT and 30 were randomly selected to increase diversity.

To evaluate every of the 8 combination of methods, considering the 100 news headlines that were given as input, means that there were 800 expressions to be evaluated. A total of 20 different surveys – each with 5 headlines and its associated generated expressions – were automatically created using Google Forms and Google Developers (using Google Apps Script[2], through a *Javascript* based API). As Google Forms does not allow randomizing questions through different forms, there were 20 different links to be shared. This problem was solved through the creation and utilization of a Department of Informatics Engineering (DEI) Cloud's Virtual Machine. Using a Ubuntu operating system, Apache2[3] and PHP 7.4[4], a server was developed under the name of `http://falaremproverbios.dei.uc.pt`. This page's only function was to distribute the people wanting to answer a survey to a randomly selected form from the set of 20. This way, each answer submission answered questions regarding 5 news headlines, for a set of 40 expressions to be analysed in each survey. One hundred survey answers were submitted by a group of several human judges, who were instructed on what was the goal of the outputs with regard to the input's context. An example of the surveys is depicted in Figure 6.2.



Considerando o título de notícia: "Noivos desesperam por terminar lua de mel, mesmo estando "presos" nas Maldivas"

- Relacionamento: até que ponto a expressão pode estar relacionada com o título de notícia anterior e ser até usado como um sub-título ou comentário;
  1- Nada relacionado;
  2- Algo relacionado;
  3- Relacionado.
- Originalidade: até que ponto a expressão lhe é familiar ou uma novidade;
  1- Já conhecia;
  2- Faz-me lembrar uma expressão conhecida, mas tem pequenas alterações;
  3- Nova para mim.
- Piada: até que ponto a expressão é engraçada;
  1- Nada engraçada;
  2- Consigo pensar num cenário em que pode ter piada;
  3- Tem piada.

Classifique a expressão: "na terra de fim e mel" *

	1	2	3
Relacionamento	○	○	○
Originalidade	○	○	○
Piada	○	○	○

Figure 6.2: Example of the public survey used for evaluating the final results..

For each headline, judges were presented with the headline followed by the list of texts by each combination and were asked to use a 3-point Likert scale for ranking, considering three different key aspects the text should exhibit for achieving the goals set: (i) relatedness; (ii) originality; (iii) funniness. Due to syntax's good score in the preliminary evaluation, it was replaced by a new aspect. Syntax is the less subjective aspect of the ones considered in

[2] `https://www.google.com/script/start/`
[3] `https://httpd.apache.org/`
[4] `https://www.php.net/`

this work, as no expressions were classified with 1 in the preliminary evaluation, regarding this aspect.

As a reminder, the **relatedness** aspect aims to classify the relation between the expression and the input in terms of semantical context. In regard to this aspect, volunteers were asked to classify resulting expressions as follows:

1. There is minimal or no relation at all between the generated expression and the input;
2. The expression is somewhat related to the input, as it shares one or two words and its context is somewhat relatable to the one of the input;
3. The expression is clearly related to the input's context and may be of use as a candidate to replace the input itself or as a comment to the input.

The new aspect to be judged was **originality**, that can also be described as considering an expression in terms of familiarity and novelty. The produced text should be novel and not only reuse existing expressions, as well as be innovative in terms of its association to the headline. Originality scores aim to evaluate how familiar the judge is with the expression, e.g. if the expression if very common and can be generally used in different contexts, it is too familiar and thus not so original. However, this can be balanced if the expression is adapted in a correct and interesting manner, as it would introduce novelty, thus increasing the expression's originality. Classification was defined as:

1. Already knew the expression before reading it;
2. Reminds me of some know expression, but with some changes;
3. This expression is new to me.

With regard to **funniness**, it is important to re-emphasize its inherent subjectivity. Similarly to the preliminary evaluation, judges did not have to justify why they found the expression funny or not funny, as to avoid influencing their opinions in terms of score. To classify funniness, the 3-point scale had these points:

1. The expression is not funny and will most likely not make anyone laugh;
2. The expression is somewhat funny and could be potentially be funny, depending on the reader's subjective view;
3. The expression is very funny, and possesses a great potential to make other people laugh.

For a first analysis of the evaluation's results, Table 6.1 presents the percentages of the scores (1, 2, and 3) with regard to the total number of answers for each of the possible combination of methods. These combinations include all six pairs of generation plus selection methods, and using only the two selection methods as well. For easier comparison between generation methods and between selection methods, the total for each approach was also computed. In terms of relatedness, the scores were evenly distributed, without any noticeable tendency, as each method has a different balance of scores. It is also important to notice that relatedness is improved when the adaptation methods are considered, thus supporting the hypothesis that the adapted texts are more closely related to the inputs than the original expressions. In fact, the only situation where using the selection algorithm, without previous adaption, has higher score than the selection algorithm's total score is for BERT, when considering funniness. Other than that, the selection algorithms' totals are always higher than their solo usage, thus supporting the use of the adaptation methods.

Concerning the maximum score, originality had the highest number of answers equal to 3, with funniness being the lowest, with some combinations having less than 20% of answers with the top score. Moreover, in the funniness aspect a percentage of over 50% answers equal to 1 was recorded in some combinations, reflecting the difficulty in generating humoristic content.

Method	**Relatedness (%)**			**Originality (%)**			**Funniness (%)**			**Total**
	1	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**Evals**
Subs+TFIDF	25.8	38.0	36.2	17.6	40.8	41.6	45.2	29.4	25.4	500
Subs+BERT	44.8	34.0	21.2	22.0	36.4	41.6	49.2	30.6	20.2	500
Total Subs	35.3	36.0	28.7	19.8	38.6	41.6	47.2	30.0	22.8	1,000
Analogy+TFIDF	29.0	37.8	33.2	17.4	36.4	46.2	44.6	29.8	25.6	500
Analogy+BERT	38.8	36.6	24.6	20.0	33.8	46.2	52.0	28.2	19,8	500
Total Analogy	33.9	37.2	28.9	18.7	35.1	46.2	48.3	29.0	22.7	1,000
VecDiff+TFIDF	34.6	35.6	29.8	17.0	35.4	47.6	53.4	27.0	19.6	500
VecDiff+BERT	38.4	35.6	26.0	16.4	32.2	51.4	51,8	24,2	24,0	500
Total VecDiff	36.5	35.6	27.9	16.7	33.8	49.5	52.6	25.6	21.8	1,000
Sel. TF-IDF	38.4	32.0	29.6	33.8	34.6	31.6	53.0	27.8	19,2	500
Total TF-IDF	31.9	35.9	32.2	21.4	36.8	41.8	49.0	28.5	22.5	2,000
Sel. BERT	52.5	27.8	19.8	22.0	35.0	43.0	46.0	29.8	24.2	500
Total BERT	43.6	33.5	22.9	20.1	34.3	45.6	49.7	28.2	22.1	2,000

Table 6.1: Proportions of the final evaluation's scores.

Table 6.2 shows several figures of the final evaluation. As previously mentioned, eight combinations of generation and selection methods were evaluated. In addition, five more combinations were added for statistical purposes, by combining a total for each method, both selection and generation, thus presenting independent results between them as well. The metrics used to evaluate the results were the mode (Mo), means (μ) and standard deviation (σ), and median (Md) of the scores for texts produced by each combination of methods. For some initial conclusions, the results show that the aspect with globally higher scores was originality, followed by relatedness and the funniness with lower scores.

Method	**Relatedness**			**Originality**			**Funniness**		
	Mo	$\mu \pm \sigma$	**Md**	**Mo**	$\mu \pm \sigma$	**Md**	**Mo**	$\mu \pm \sigma$	**Md**
Subs+TFIDF	2	2.10 ± 0.78	2	3	2.24 ± 0.73	2	1	1.80 ± 0.82	2
Subs+BERT	1	1.76 ± 0.78	2	3	2.20 ± 0.77	2	1	1.71 ± 0.78	1
Total Subs	1	1.93 ± 0.80	2	3	2.21 ± 0.75	2	1	1.76 ± 0.80	2
Analogy+TFIDF	2	2.04 ± 0.79	2	3	2.23 ± 0.74	2	1	1.81 ± 0.82	2
Analogy+BERT	1	1.86 ± 0.78	2	3	2.26 ± 0.77	2	1	1.68 ± 0.78	1
Total Analogy	2	1.95 ± 0.80	2	3	2.25 ± 0.76	2	1	1.74 ± 0.80	1
VecDiff+TFIDF	1	1.95 ± 0.80	2	3	2.30 ± 0.74	2	1	1.66 ± 0.79	1
VecDiff+BERT	1	1.88 ± 0.80	2	3	2.35 ± 0.75	3	1	1.72 ± 0.83	1
Total VecDiff	1	1.91 ± 0.80	2	3	2.33 ± 0.75	2	1	1.69 ± 0.81	1
Sel. TF-IDF	1	1.91 ± 0.82	2	2	1.98 ± 0.81	2	1	1.66 ± 0.78	1
Total TF-IDF	2	2.00 ± 0.80	2	3	2.20 ± 0.77	2	1	1.73 ± 0.80	2
Sel. BERT	1	1.67 ± 0.78	1	3	2.21 ± 0.78	2	1	1.78 ± 0.81	2
Total BERT	1	1.79 ± 0.79	2	3	2.26 ± 0.77	2	1	1.72 ± 0.80	2

Table 6.2: Evaluation figures.

To measure the agreement between evaluators, the Fleiss coefficient was used. As said before, this metric is similar to Cohen's Kappa coefficient, but is able to measure the reliability of agreement between more than two evaluators. As there were 20 different surveys with multiple evaluators each, the metric was computed for each survey and each validated aspect: (i) Relatedness agreement was equal to 0.051; (ii) originality agreement was equal to −0.014; (iii) funniness agreement was equal to 0.036. To interpret these results, it possible to use Cohen's Table 4.3. The metric shows that there is just a slight agreement in terms of relatedness and funniness, and a poor agreement in terms of originality, enhancing the hypothesis that this kind of content and evaluation is highly subjective. Another problem might have been a lack of understanding from the evaluators, either because they did not understand the scope of the work, or because they were unable to identify obvious relations. This might have influenced the high originality scores, as the evaluators might have analysed the originality of the association between input and output, or just did not know many expressions to begin with. This is always a risk when validating results through the opinions of humans.

Some of the best expressions for each of the evaluated aspects are presented in Tables 6.3, 6.4, and 6.5. These Tables present, for each expression: (i) the combination of methods that originated the expression; (ii) the headline given as input; (iii) the original expression; (iv) the output; (v) the means of all three aspects: relatedness, originality and funniness, respectively.

Relatedness

By looking at the relatedness results, the proportions between ratings are mostly all evenly distributed. However, the combinations' mode is mainly valued at 1 and there is no significant difference between the methods, even considering the means. Even though there are slight differences in terms of means, the large standard deviation values do not allow many conclusions to be drawn. However, it is possible to define the best two combinations, in this aspect, which are the only two whose mode is 2: Substitution + TF-IDF and Analogy + TF-IDF. This is also supported as these two combinations, in terms of relatedness, are the only two whose percentage of expressions evaluated with 3 is higher than those evaluated with 1. In addition, Vector Difference + TF-IDF is another combination whose mean almost reaches 2.00, while the other combinations fall shorter.

From this, it is possible to determine that the TF-IDF is the selection method with higher scores in terms of relatedness, which was supported by its total scores, whose mode was also 2, contrasting BERT's mode of only 1. It is also important to notice a repetition of results from the evaluation presented in Section 4.3, in the combinations that disregarded generation methods, opting for using only the selectors. In these combinations, TF-IDF also produced better results than BERT, with the combination that only used BERT selection being the only one whose median was less than 2. This may be due to TF-IDF's ability to search for expressions, generated or unaltered, that share some common words with the input, thus possibly making the expression more relatable. This effect is easily seen in the results presented in Table 6.3, where the replaced and substituted words are underlined, usually with the substitute being directly imported from the input. This is not necessarily true in other cases, like the first example in Table 6.4, but since these were some of the outputs that had better evaluations in terms of relatedness, sharing words with the inputs seems the probable cause.

The expressions that received better evaluations in terms of relatedness are depicted in Table 6.3. These examples show that, for this aspect, the TF-IDF selection method produces outputs that are more easy to relate, as they usually share more words with the input than the outputs selected by BERT. With regard to the generation methods, there is not a distinguished method, as they all had similar scores. For better understanding of the process under which the original expressions were submitted, the words to be replaced and their respective substitutes are underlined.

Method	Input	Original	Output	μ

Analogy + TF-IDF	*Rooney sobre cortes de salários: 'Porque é que são os futebolistas os bodes expiatórios?'* (Rooney about salary cuts: 'Why are footballers the scapegoats?)	*Os últimos são os primeiros.* (Those who finish last, are the first.)	*Os expiatórios são os primeiros.* (Scapegoats are the first.)	2.75;
				2;
				2.5;

Subs + TF-IDF	*Bancos dizem que as condições das linhas de crédito foram definidas pelo governo* (Banks claim that credit conditions were defined by the government)	*Dar a César o que é de César e a Deus o que é de Deus* (To Caesar what is Caesar's and to God what is God's)	*Dar a governo o que é de governo e a Deus o que é de Deus* (To the government what is the government's and to God what is God's)	3;
				2;
				2.5;

VecDiff + TF-IDF	*Homem barrica-se após atacar uma empregada de limpeza com catana em Lisboa* (Man hides after attacking a cleaning lady with sword in Lisbon)	*Um Homem Com Sorte* (A man with luck.)	*Um homem com catana* (A man with sword)	2.8;
				2;
				1.4;

Table 6.3: Examples of the best evaluated expressions considering relatedness.

Originality

In terms of the originality aspect, most of the combinations had mode equal to 3, which is a very positive result, as 3 is the maximum score. This is also supported by the fact that almost every combination has more than 40% of expressions with rating equal to 3. Even though the results were good, once again there were not any notorious differences between combinations, as they have very similar means and large standard deviation numbers. However, there is one generation method that has a slightly better score, which is Vector Difference. Any of this method's combinations has a mean score of over 2.30, which is greater than any other combination. In addition, the combination Vector Difference + BERT was the only combination that achieved a median of 3, thus proving some superiority in the originality aspect. The reason that supports these results is that Vector Difference is the generation method that is able to replace two words for each adapted expression with higher success rate, like in the third example of Table 6.5. This causes the output to be a novelty more often, as the expressions suffer from more adaptations, thus being less familiar to the reader.

Considering the selection methods, the total scores value BERT as the most original, with the combination of only using the TF-IDF selector being the only one whose mean value is under 2.00. This lowered the method's total score, which was lower than BERT's, but otherwise, there is no distinguishable difference between the selectors when used as a complement to the generation methods. This depreciation of the TF-IDF selector is probably due to the same reason its relatedness scores are higher: as it uses expressions that most likely share common words with the input, the probability of choosing popular sayings, full of familiar and common words, is higher. This supports the lower scores of using only the TF-IDF selector.

A group of expressions with high originality scores are gathered in Table 6.4. There is an example for each generation method, and words to be replaced and their substitutes are also underlined for better understanding. In this aspect, BERT provides a possibly better selection, as its choices do not rely so much on shared words, but mostly on the full context of the considered expressions. As such, it is much more likely to select expressions whose familiarity is lower and novelty higher, as they do not necessarily have an immediate and obvious connection to the input, even though their contexts are intertwined.

Method	Input	Original	Output	μ

Subs + BERT	*Património Cultural propõe classificação do Mosteiro de Coz como monumento nacional* (Cultural heritage proposes a national monument classification to Mosteiro de Coz)	*O rapaz e o monstro* (The boy and the monster)	*O rapaz e o túmulo* (The boy and the tomb)	1.2;
				2.8;
				1;

Analogy + BERT	*Hidroxicloroquina. Trump toma, autoridades não recomendam e opositores criticam* (Hydroxychloroquine. Trump takes, officials do not recommend and opponents criticize)	*Quem não aparece, esquece* (He who does not show up, forgets)	*Quem não recomenda, aponta* (He who does not recommend, points)	2.25;
				3;
				1.25;

VecDif + BERT	*GNR descontamina ambulâncias de todo o país.* (GNR decontaminates ambulances across the country)	*Valete: Baile das Máscaras* (Jack: The Masquerade Ball)	*Valete: País das Máscaras.* (Jack: Country of Masks)	1.6;
				3;
				1.2;

Table 6.4: Examples of the best evaluated expressions considering originality.

Funniness

Considering the funniness aspect, it was predictable that it would have lower scores, for its subjectivity and also for the difficulty of some themes presented in the inputs, e.g. it is not easy or adequate to find expressions funny when the headlines context is a medical emergency or some tragedy, except in what is considered to be dark humour. With such dependency on the context, it is not easy to assure funny results. This is reflected in the fact that no combination has a mode of more than 1, meaning that no combination was very successful in this aspect. However, there were positive signs, like two combinations of methods with median scores equal to two, enabling the determination of which combinations outproduced the others in this regard: Substitution + TF-IDF, Analogy + TF-IDF, and using only the BERT selector. A possible reason for the better results of these generation methods when compared to Vector Difference may be that the latter forcefully reuses the words from the input as replacement material for the expression adaptation. On the other hand, Substitution and Analogy allow the use of words that are not present in neither the input or the saying, leading to the possible appearance of more unexpected words that may force some incongruence within the sentence, thus giving it more humoristic value.

It is very difficult to distinguish the selector methods in terms of funniness, as TF-IDF performs better with the aid of a generation method – especially Substitution and Analogy – while BERT clearly has a better performance on its own, even reaching a median of 2, as was said before. In summary, both of the selectors totals are almost the same with regard to funniness.

Table 6.5 demonstrates some successful examples in terms of funniness. Considering the generated expression *"Sorte no jogo, peixe no amor"*, its funniness derives from the replacement of *"azar"* by *"peixe"*, which generates an incongruence as fish is not usually associated with love, thus making it potentially funny. In the second example, applying that output to that headline also means calling a group of humans, subjected to a suspended droplet, a sheep herd, which as realistic and relatable as it is funny (see the semantic constraints in the work of Valitutti et al. [2016], described in Chapter 4). In the third example, as the Vector Difference method replaced two words from the original saying with two words from the input, it generated an expression that demonstrates the desperation of the headline's referenced newlyweds while giving it a touch of humour through the usage of the word *"mel"* (honey).

Method	Input	Original	Output	μ

Subs + TF-IDF	*Quebra de preços no peixe preocupa os pescadores* (Fish price drop worries fishermen)	*Sorte no jogo, azar no amor* (Good luck in the game, bad luck in love)	*Sorte no jogo, peixe no amor* (Good luck in the game, fish in love)	1.57;
				2.29;
				2.57;

Analogy + BERT	*Uma simples conversa gera gotículas que podem ficar suspensas no ar até 14 minutos* (A simple conversation generates droplets that can be suspended in the air for up to 14 minutes)	*Uma ovelha má põe o rebanho a perder.* (A bad sheep causes the herd to lose)	*Uma gotícula suspensa põe o rebanho a perder* (A suspended droplet causes the herd to lose)	2.25;
				2.75;
				2.5;

VecDif + BERT	*Noivos desesperam por terminar lua de mel, mesmo estando "presos" nas Maldivas* (Newlyweds are desperate to end their honeymoon, even though they are "stuck" in the Maldives)	*Ouro é o que ouro vale* (Gold is what gold is worth)	*Mel desespera o que mel vale* (Honey despairs what honey is worth)	2.25;
				2.5;
				2;

Table 6.5: Examples of the best evaluated expressions considering funniness.

6.3 Twitter bot *@TextoEmContexto*

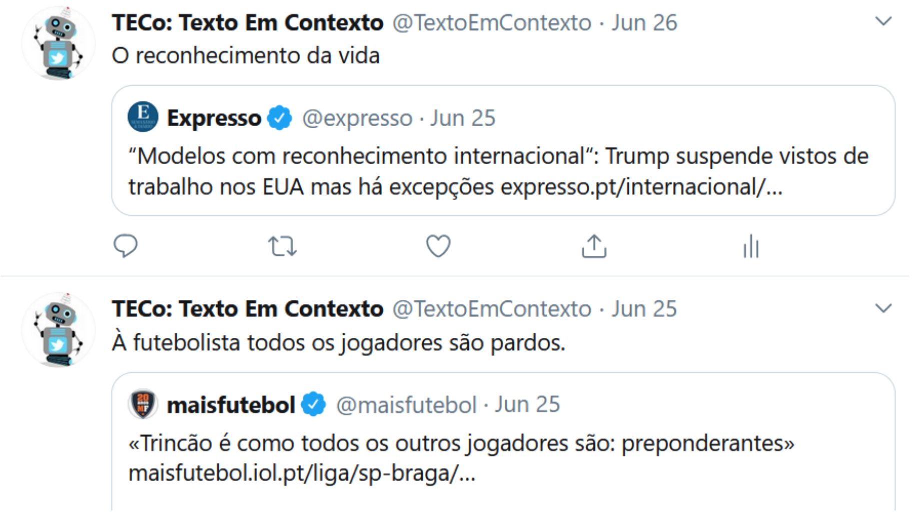
The first step of developing a bot was to create a Twitter Developer account, which allows the programmer to develop applications whose objectives may be very diverse, as to publish Tweets and interact with other accounts like an ordinary user's account, retrieve data for posterior analysis, optimize ads, within others. To facilitate the retrieval of data, the bot followed several known Portuguese newspapers and serious public figures, e.g. the Prime-Minister of Portugal, in order to collect several headlines, either from news articles or formal tweets.

A Twitter developer account makes it possible to manage the public account through the API provided in this platform. To aid the development of the bot, a Python library, named Tweepy[5], was used as tool for accessing the Twitter API through OAuth authentication, a pattern used to authorize third parties to login and use accounts. Afterwards, from the account's home timeline the bot is able to retrieve a number of Tweets, which were already valid in terms of context, as the account only followed a constrained set of other accounts. The bot is then able to output a list of tweets, which were composed of news headlines, as a text file to be used as input by the system.

In addition to these data retrieval capabilities, the bot, named *@TextoEmContexto*, is also able to post tweets of its own and even post as Retweet with comment, which is a tweet whose text also has a some other tweet's content, like the examples depicted in Figure 6.3.

At the moment, the bot works at command, not running constantly. However, it will be uploaded to the mentioned Virtual Machine (see Section 6.2), as a way to be running constantly, intervening at a predefined time interval. At each given time, the bot checks the account's home timeline through the API's *home_timeline()* method that allows the extraction of tweets. Tweets are retrieved as dictionaries with a lot of data, but for this work's purpose, only the tweet itself and the id of the account who posted it are saved. From here, TECo is called, receiving the tweet as input. For automation purposes, the

[5] `https://www.tweepy.org/`



TECo: Texto Em Contexto @TextoEmContexto · Jun 26

O reconhecimento da vida

Expresso @expresso · Jun 25

"Modelos com reconhecimento internacional": Trump suspende vistos de trabalho nos EUA mas há excepções expresso.pt/internacional/...

TECo: Texto Em Contexto @TextoEmContexto · Jun 25

À futebolista todos os jogadores são pardos.

maisfutebol @maisfutebol · Jun 25

«Trincão é como todos os outros jogadores são: preponderantes» maisfutebol.iol.pt/liga/sp-braga/...

Figure 6.3: Two examples of tweets posts by the Twitter bot *@TextoEmContexto*.

generation and selection methods used by the bot are being chosen at random, but there is also the possibility of running each combination of methods and selecting the best option from the total of the generated expressions. After the output is generated, a new tweet is build with the output's string and the id of the original tweet, as the bot retweets the original tweet with a commentary, which is the generated expression.

For future endeavours, it will also be possible to retrieve several tweets and choose between them by popularity – measured in Twitter by the number of *reposts* and *likes* – to increase the popularity of the bot and guarantee inputs with quality.

6.4 Discussion

In this chapter, the fully developed TECo system was presented, including its architecture in Section 6.1, that englobes the selection methods seen in Chapter 4 and the adaptation methods presented in Chapter 5. This chapter concludes by describing and presenting the final evaluation of the system's results, both with statistical data and examples of generated outputs, separated by their respective aspect, namely: (i) relatedness; (ii) originality; (iii) funniness.

The results were not as high as it could be expected, since this kind of creative text is very subjective, particularly when it comes to humour. However, some good indications were possible to be drawn from the results, which could be used in future contributions. For instance, every evaluated aspect increases with the adapted texts when compared to using only selection approaches to original texts.

Concerning relatedness, there is not much difference between generation methods, even though there is a slight advantage tending towards Substitution and Analogy, when combined with TF-IDF selector. Considering the selectors, TF-IDF proved to be more reliable, probably due to its ability to search for expressions that share common words with the input.

Originality scores were very positive, again with no significant differences between methods, but with a slight tendency for Vector Difference. This method, being the one with most success replacing more than one word per expression, is more unpredictable than the other two, thus possess higher probability of producing novel content. In this aspect, the

BERT selector had better results, also due to its unpredictability, having higher probability of selecting less known expressions whose association to the input may not be immediate, and thus increasing their novelty and, consequently, their originality scores.

Funniness was predicted to have the lower scores, both for its highly subjective content, and because it was not the main goal of this work. This aspect is also very influenced by the headlines' context, as some contexts do not allow much room for funniness. However, there were two combinations with better scores, Substitution and Analogy, when paired with TF-IDF selector. These two combinations were also the ones with better scores in terms of relatedness. As for the selectors, using only BERT proved to be successful, as its unpredicability may select unexpected outputs, whose association to the input may provoke laughter.

Finally, the presentation of the Twitter bot *@TextoEmContexto* – that can be used for data retrieval and for posting outputs – in Section 6.3. This bot was used mainly to fetch news headlines, but in the future it will be automated to run constantly and post TECo's results from time to time.

Chapter 7

Conclusion

This thesis is included in the scope of Computational Creativity (CC), more specifically in the sub-field of Linguistic creativity, which is an area with a development spectrum of many possibilities, and is becoming more important everyday. The idea of developing a system that handles Portuguese text is natural, as it is the author's mother tongue, and the inherent challenge due to the existence of fewer systems and resources dedicated to this language. In addition, Portuguese culture has a marked presence in its language and its expressions, whose value is intangible.

The system developed in the scope of this thesis, named Texto Em Contexto (TECo) (Text in Context), has two main goals: (i) selecting expressions based on the context of a given input (see Chapter 4); (ii) generating novel expressions through proposed adaptation methods to approximate the expression to the input's context (see Chapter 5).

Several concepts were required, described in Chapter 2, including: (i) the definition of humour (Section 2.1); (ii) the definition of CC (Section 2.2) and how it can be applied to develop creative and autonomous systems; (iii) methodologies present in Natural Language Processing (NLP), with emphasis on morphology and semantics (Section 2.3); (iv) concepts regarding Distributional Semantics (Section 2.4, from word representations like Word Embeddings (WEs), to semantic similarity metric like Cosine Similarity (CS), to metrics like the Term Frequency - Inverse Document Frequency (TF-IDF) algorithm, and to state-of-the-art approaches like transformers and, more specifically, Bidirectional Encoder Representations from Transformers (BERT).

The development of this system was influenced by various related works from other authors, presented in Chapter 3. These works were chosen due to their similarities to TECo, describing many examples that also mix analysis of both morphology and semantics, in addition to some that even apply linguistic generation.

Concerning the first main goal of this thesis, a sub-system, also called the short-text selector based on context, was presented in Chapter 4, where the different approaches used for selection and their implementation were explained. An evaluation and further discussion was made for the results of this sub-system, as a full paper describing it was accepted at the International Conference on Computational Creativity (ICCC). Chapter 5 presents the second sub-system, proposing three methods for textual generation: (i) Substitution; (ii) Analogy; (iii) Vector Difference. Despite their similar structures, they differ in the approach to choose and replace words in the given expressions. Furthermore, their implementation was also presented, along with an experimental setup designed to permit a preliminary validation of results to validate the methods, identify, and fix minor issues.

Finally, Chapter 6 presents the fully developed TECo system, englobing both subsystems. The full system architecture was presented, before diving into the systematic evaluation made by several human judges, along with its results and further discussion on this content. The system's outputs were evaluated in three aspects: (i) relatedness; (ii) originality; (iii) funniness. Despite the risk inherent to human subjectivity, it was possible to draw some conclusions regarding the system's results. For instance, each validated aspect increases its scores using any adaptation methods, when compared to only using selectors on original expressions. Both relatedness and originality scores show promising results, as the outputs were considered to be generally more related to the inputs, while achieving a good degree of novelty, a determinant aspect for a creative system. However, with regard to funniness, the results were not as good. Humour was not directly tackled in this work, and its high subjective value, either due to the opinion of the reader, or the input's context, hampered the system's results.

Future Endeavours

For future endeavours, there are several approaches that could be tackled. Concerning selection methods, specifically BERT, may be improved through the process of fine-tuning. Furthermore, there are other options to encode full sentences with BERT, instead of encoding token by token and averaging their representations. As said before, humour is not the main focus of this thesis, being an interesting aspect to evaluate due to the work's use of figurative language and subsequent possibility of inducing humour. However, it would be interesting to develop a classifier to rank the approaches, like the one developed by Clemêncio et al. [2019].

Considering TECo as a full system, it may be integrated in a application such as a chat-bot, where the bot would respond with suggestions of expressions to the user's input. This could also be used as a tool to inspire more creative texts.

References

E. Agirre, D. Cer, M. Diab, and A. Gonzalez-Agirre. Semeval-2012 task 6: A pilot on semantic textual similarity. In ** SEM 2012: The First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393, 2012.

Y. Ahn, H. Lee, H. Jeon, S. Ha, and S. G. Lee. Quote recommendation for dialogs and writings. *CEUR Workshop Proceedings*, 1673:39–42, 2016. ISSN 16130073.

K. Alnajjar and H. Toivonen. Computational generation of slogans. *Natural Language Engineering*, pages 1–33, 2020.

K. Alnajjar, L. Leppänen, and H. Toivonen. No time like the present: Methods for generating colourful and factual multilingual news headlines. In *Proceedings of 10th International Conference on Computational Creativity (ICCC)*, pages 258–265. Association for Computational Creativity, 2019.

D. Ameixa, L. Coheur, P. Fialho, and P. Quaresma. Luke, i am your father: dealing with out-of-domain requests by using movies subtitles. In *International Conference on Intelligent Virtual Agents*, pages 13–21. Springer, 2014.

C. Aouameur, P. Esling, and G. Hadjeres. Neural drum machine: An interactive system for real-time synthesis of drum sounds. In K. Grace, M. Cook, D. Ventura, and M. L. Maher, editors, *10th International Conference on Computational Creativity*, chapter 3, pages 92–99. Mouton de Gruyter, 2019.

S. Attardo. A primer for the linguistics of humor. In V. Raskin and W. Ruch, editors, *The Primer of Humor Research*, chapter 3, pages 110–156. Mouton de Gruyter, 2008.

F. Barron and D. M. Harrington. Creativity, intelligence, and personality. *Annual review of psychology*, 32(1):439–476, 1981.

B. Bay, P. Bodily, and D. Ventura. Text transformation via constraints and word embedding. In *ICCC*, pages 49–56, 2017.

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

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

A. Clemêncio, A. Alves, and H. Gonçalo Oliveira. Recognizing humor in Portuguese: First steps. In *Proceedings of 19th EPIA Conference on Artificial Intelligence, EPIA 2019,* *Vila Real, Portugal, September 3-6, 2019, Part II*, volume 11805 of *LNCS/LNAI*, pages 744–756. Springer, September 2019.

S. Colton and G. A. Wiggins. Computational creativity: The final frontier? In *Proceedings of 20th European Conference on Artificial Intelligence (ECAI 2012)*, volume 242 of *Frontiers in Artificial Intelligence and Applications*, pages 21–26, Montpellier, France, 2012. IOS Press.

J. M. Cunha, P. Martins, H. Gonçalo Oliveira, and P. Machado. Ever-changing flags: Trend-driven symbols of identity. *8th Conference on Computation, Communication, Aesthetics & X (xCoAx)*, 2020.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

A. Drozd, A. Gladkova, and S. Matsuoka. Word embeddings, analogies, and machine learning: Beyond king - man + woman = queen. In *Proceedings the 26th International Conference on Computational Linguistics: Technical papers (COLING 2016)*, COLING 2016, pages 3519–3530, 2016.

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

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

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

L. Gatti, G. Özbal, M. Guerini, O. Stock, and C. Strapparava. Slogans are not forever: Adapting linguistic expressions to the news. *IJCAI International Joint Conference on Artificial Intelligence*, 2015-Janua(Ijcai):2452–2458, 2015. ISSN 10450823.

P. Gervás. Computational approaches to storytelling and creativity. *AI Magazine*, 30: 49–62, Jul. 2009. doi: 10.1609/aimag.v30i3.2250. URL `https://www.aaai.org/ojs/index.php/aimagazine/article/view/2250`.

P. Gervás. Generating a search space of typical narrative plots. In K. Grace, M. Cook, D. Ventura, and M. L. Maher, editors, *10th International Conference on Computational Creativity*, chapter 7, pages 228–235. Mouton de Gruyter, 2019.

H. Gonçalo Oliveira. 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)*, pages 11–20, Santiago de Compostela, Spain, September 2017. ACL Press.

H. Gonçalo Oliveira and R. Rodrigues. Exploring lexical-semantic knowledge in the generation of novel riddles in Portuguese. In *Proceedings of the 3rd Workshop on Computational Creativity in Natural Language Generation*, CC-NLG 2018, pages 17–25, Tilburg, The Netherlands, November 2018. ACL Press.

H. Gonçalo Oliveira, D. Costa, and A. Pinto. 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, pages 238–245, Paris, France, 2016.

Z. S. Harris. Distributional structure. *WORD*, 10(2-3):146–162, 1954. doi: 10.1080/00437956.1954.11659520.

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

S. Hawking. Stephen Hawking warns artificial intelligence could end mankind - BBC News, 2014. URL `https://www.bbc.com/news/technology-30290540`.

D. Jurafsky and J. H. Martin. *Speech and Language Processing (3rd Edition)*. Draft chapters available from `https://web.stanford.edu/~jurafsky/slp3/`, 2019.

D. Kim, J. Xu, A. Elgammal, and M. Mazzone. Computational analysis of content in fine art paintings. In K. Grace, M. Cook, D. Ventura, and M. L. Maher, editors, *10th International Conference on Computational Creativity*, chapter 2, pages 33–40. Mouton de Gruyter, 2019.

J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, 1977. ISSN 0006341X, 15410420. URL `http://www.jstor.org/stable/2529310`.

M. Loller-Andersen and B. Gambäck. Deep learning-based poetry generation given visual input. In F. Pachet, A. Jordanous, and C. León, editors, *9th International Conference on Computational Creativity*, pages 240–255. Mouton de Gruyter, 2019.

E. Loper and S. Bird. 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*, pages 63–70, 2002.

R. Martin and T. Ford. *The Psychology of Humor: An Integrative Approach.* Elsevier Science, 2018. ISBN 9780128135099.

R. Mendes and H. Gonçalo Oliveira. Comparing different methods for assigning portuguese proverbs to news headlines. In *Procs. of 11th ICCC*, page Accepted for publication, 2020.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, pages 1–12, 2013a.

T. Mikolov, W. Yih, and G. Zweig. Linguistic Regularities in Continuous Space Word Representations - Microsoft Research. *Hlt-Naacl*, pages 746–751, 2013b.

M. D. Mumford. Where Have We Been, Where Are We Going? Taking Stock in Creativity Research. *Creativity Research Journal*, 15(2-3):107–120, jul 2003. ISSN 1040-0419. doi: 10.1080/10400419.2003.9651403. URL `http://www.tandfonline.com/doi/abs/10.1080/10400419.2003.9651403`.

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

J. Pennington, R. Socher, and C. Manning. 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, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL `https://www.aclweb.org/anthology/D14-1162`.

E. Ranchhod, C. Mota, and J. Baptista. A computational lexicon of Portuguese for automatic text parsing. In *Procs of SIGLEX99 Workshop: Standardizing Lexical Resources*. ACL Press, 1999.

V. Raskin. *The primer of humor research*. Mouton de Gruyter, 2008. ISBN 978-3-11-019849-2.

P. A. Rocha and D. Santos. CETEMPúblico: Um corpus de grandes dimensões de linguagem jornalística portuguesa. In *V Encontro para o processamento computacional da língua portuguesa escrita e falada (PROPOR 2000)*, pages 131–140, São Paulo, 2000. ICMC/USP.

A. Rodrigues, A. Cardoso, and P. Machado. A dynamic approach for the generation of perceptual associations. In K. Grace, M. Cook, D. Ventura, and M. L. Maher, editors, *10th International Conference on Computational Creativity*, chapter 10, pages 301–305. Mouton de Gruyter, 2019.

E. K. Shimomoto, L. S. Souza, B. B. Gatto, and K. Fukui. News2meme: An automatic content generator from news based on word subspaces from text and image. In *2019 16th International Conference on Machine Vision Applications (MVA)*, pages 1–6, 2019.

T. Sousa, H. Gonçalo Oliveira, and A. Alves. Assessing different methods for solving analogies with portuguese word embeddings. In *Proceedings of 9th Symposium on Languages, Applications and Technologies (SLATE 2019)*, OASIcs, page In press. Schloss Dagstuhl, July 2020.

O. Stock and C. Strapparava. Hahacronym: A computational humor system. In *Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*, pages 113–116. Association for Computational Linguistics, 2005.

S. E. O. Tagnin. O humor como quebra da convencionalidade. *Revista Brasileira de Lingüística Aplicada*, 5(1):247–257, 2005.

A. Valitutti, A. Doucet, J. M. Toivanen, and H. Toivonen. Computational generation and dissection of lexical replacement humor. *Natural Language Engineering*, 22(5):727–749, 2016. ISSN 14698110. doi: 10.1017/S1351324915000145.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017-December(Nips):5999–6009, 2017. ISSN 10495258.

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

T. Veale, K. Feyaerts, and C. Forceville. 1. Creativity and the Agile Mind. *Creativity and the Agile Mind*, pages 1–24, 2013. doi: 10.1515/9783110295290.15.

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

M. Žnidaršič, A. Cardoso, P. Gervás, P. Martins, R. Hervás, A. Alves, H. Gonçalo Oliveira, P. Xiao, S. Linkola, H. Toivonen, J. Kranjc, and N. Lavrač. Computational creativity infrastructure for online software composition: A conceptual blending use case. In *Proceedings of 7th International Conference on Computational Creativity*, ICCC 2016, Paris, France, 2016.

V. Zoric and B. Gambäck. The image artist: Computer generated art based on musical input. In F. Pachet, A. Jordanous, and C. León, editors, *9th International Conference on Computational Creativity*, pages 296–303. Mouton de Gruyter, 2019.

Appendices

Appendix A

This appendix presents the full paper accepted in International Conference on Computational Creativity (ICCC) 2020.

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 context, such as news headlines. Given an input text, different methods are applied to select a corresponding expression, which should be as semantically related to the input as possible. This study was developed for the Portuguese language, and the suggested short texts are proverbs and sayings, i.e. for each headline a proverb is automatically selected. The set of explored methods includes measures based on word overlap, on static word embeddings and also on recent contextual embeddings based on transformers. In order to compare the explored methods in this subjective scenario, a survey was answered by humans, who rated the relatedness and humor value of proverbs selected for different headlines. The main conclusion was that simpler approaches, which end up selecting proverbs 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 the English language. The study described in this paper handles the Portuguese language which, despite a large number of speakers, has a much smaller research community, thus presenting different challenges. We propose an automatic selector of proverbs able to, given a news headline, choose the proverbs in relation to an input's context. For this purpose, we test and analyze diverse approaches, and see many possible and different applications. For instance, in the domain of journalism, proverbs may contribute to more appealing, creative and memorable headlines, thus increasing the number of readers; or in the domain of chatbots, using proverbs and sayings in the appropriate contexts could make conversations more interesting.

The use of Portuguese proverbs and sayings is due to their presence on the quotidian of most Portuguese people, as a way to put emphasis or make an analogy on a given situation or occurrence, usually implying humoristic value to the situation. This humoristic connotation of the proverbs makes them an interesting element to increase the appeal of a news headline. However, for a computer, it is very troublesome to find the underlying meaning beneath the use of proverbs, since they generally resort to figurative language and are thus 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, i.e. a news headline, and a list of proverbs and sayings with which to compare the input. Sometimes, proverbs are actually used as news headlines, but as a result of manual human effort, which requires a deep knowledge of popular culture and proverbs, e.g. the headline *"Burro Velho não aprende línguas, mas mata a fome a quem aparecer"* [1] ("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, so that it can be used by anyone.

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 TF-IDF algorithm, to other more recent approaches such as those based on static word embeddings. Moreover, state-of-the-art methods based on transformers are also analyzed, namely the Bidirectional Encoder Representations from Transformers (Devlin et al., 2018), which has been getting a lot of attention due to having unexpectedly good results in many 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 a section introducing background knowledge for 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 will describe how the selected methods were evaluated in the proposed scenario and discuss our interpretation of the evaluation's results.

[1] https://www.jornaldeleiria.pt/noticia/burro-velho-nao-aprende-linguas-mas-mata-a-fome-a-quem-aparecer-5284

Background Knowledge

Before applying methods for natural language processing and understanding, the input often needs to be preprocessed. Each sentence is submitted to a morphological analysis consisting of tokenization, lemmatization and Part-of-Speech (PoS) tagging, so that each word and its derivatives are considered, eliminating words whose contribution to the semantic value of the sentence is limited, such as *stop-words*, 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.

Semantic similarity The concept of **similarity** grounds itself in the meaning of the considered content. Knowing "how similar two words are can help in computing how similar the meaning of two phrases or sentences are" (Jurafsky and Martin, 2019). Semantic similarity can be a tool for computing how similar two words are, and can be estimated by different methods. There are also methods for computing the similarity between sentences, with the most simple based on averaging the values of each word. Furthermore, it is also possible to weight the values of the sentence's tokens by the use of the TF-IDF algorithm and compute the average of the tokens with this in consideration.

In the following paragraphs we will present the three simplest methods used in this study, considered traditional due to their usage before the introduction of word embeddings and other 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 as vectors of numbers, and their similarity is given by the cosine of such vectors, differing in how the vectors are computed.

CountVectorizer method (Pedregosa et al., 2011) performs tokenization on a set of text documents and constructs a vocabulary, enabling the codification of documents in regard to that vocabulary. These sparse vectors elements (as they have many elements equal to zero) represent the number of times each word appears in the set of documents.

TFIDFVectorizer bases itself on the TF-IDF algorithm, that aims to compute the relevance of a word given a certain corpus based on its frequency, thus being usually used as a weighting factor in co-occurrence matrices, which represent the amount of times, for each word, that other words appear in its context.

TF stands for Term Frequency, which is the amount of occurrences of a word w in a document, while IDF stands for Inverse Document Frequency, with document frequency being the number of documents a word appears in. This algorithm is able to reduce the weight of stop words like prepositions and determiners, that contribute little to the meaning of the text, and increase the weight of words that do not appear very often and may carry more semantic value, and are thus more relevant for discriminating between different documents.

Vector Semantics

Representing words in vectors of real numbers, also called **word embedding**, is a friendly way of computing word similarity or using words as features in a machine learning framework, creating a semantic vector space. This technique improves the larger the amount of text it relies on, as well as the diversity of the words used. 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, presenting itself as a limitation of this technique.

Word embeddings 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 far as the generation of the word embeddings is concerned, 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 to create word embeddings. This approach considers 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.

Bidirectional Encoder Representations from Transformers (BERT) A transformer (Vaswani et al., 2017) is an architecture that aims to provide a solution for converting input sequences into output sequences. BERT's architecture (Devlin et al., 2018) introduced a new way for people to work with text data, handling sequence-to-sequence tasks and long-range dependencies. It 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. This model can be fine-tuned with a new output layer in order to be adapted for a broader range of tasks, i.e. to specialize in the intended type of text, without having to suffer the impact of considerable architecture changes.

The algorithm of **self-attention** in the transformer permits 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"*.

The algorithm of self-attention starts by dividing the input sequence in three vectors: query vector, key vector and value vector. For each word *w*, the mechanism will calculate 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. As it is computed several parallel times in a Transformer's architecture, some call this algorithm Multi-head Attention.

Although BERT can be fine-tuned for computing the similarity between sequences of text, we may also use it for encoding sentences in a vector, and then compute the cosine of such vectors.

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 proposed for simply recommending quotes to be used in dialogues (Ahn et al., 2016), without much concern regarding creativity.

Considering the generation of different type of text, namely poetry, we considered the poetry generation system developed by Chrismartin and Manurung (2015) that, in addition to creating the structure of the poem, it tries to give it a pretended meaning as well. To produce text, they used a mechanism named chart generation, while capturing the semantic representations of news articles given as input from a dependency parser.

Veale, Chen, and Li (2017) explored the generation of metaphors in regard to the current news. The choice of metaphors, as well as their relation to the news, allowed the exploration of different techniques, such as Latent Dirichlet Allocation (LDA), an approach to topic modelling, the Latent Semantic Analysis (LSA), or Word2Vec. These techniques construct a joint vector space by merging the news corpus with the metaphor corpus, 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.

A multilingual system was studied by Alnajjar, Leppänen, and Toivonen (2019), who adapted an automated journalism system that generates news headlines in English and Finnish in order to increase their creativity. This system was complemented with existing tools for creative headlines, through the presentation of a suitable expression (e.g. movie title) as a catchy title, and figurative language generation, which injects similes and metaphors into headlines, according to the context of the news story, through methods of semantic similarity and classifying the prosody of the phrase and the headline, which *"is evaluated to increase the catchiness of the result"*.

Gatti et al. (2015) developed a method of generating catchy news headlines using well-known expressions which may also be used as a creativity boosting application for artists, relying on a method based on corpus of different types of text, such as cliches, movie and song titles, or slogans. For the creation of a sentence vector, the authors summed the vectors representing the words of the sentence, excluding stop words. For each slogan, the most similar headlines were selected, according to the cosine similarity metric. 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 substitute a word *w* in the slogan, considering its part-of-speech. This substitution, when successful, is then ranked with the other successful substitutions, considering the mean of its similarity and dependency scores. The selection with the lowest rank is selected and presented. Diverse outputs were returned, like one where the news headline is *"Wood: Time for Wales to step up"*, the output given by the re-analyzer is *"Unleash the power of the majority"*, while generating the expression *"Unleash the power of the sun"*.

Ahn et al. (2016) propose a system for quote recommendation in dialogues, without any creative intent. Regarding input's data, which would be short-line text, it was gathered from three different set of corpus, namely Twitter, *Project Guntenberg Database*[2] and ICWSM spinn3r blog dataset[3]. The authors propose five methods for quote recommendation, considering the pre-context (i.e., text before the quote) and post-context (i.e., text after the quote):

1. Matching Granularity Adjustment: method to measure 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: pproach that 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: using LSTM, this method consists of three parts (forget, input, output) to teach the networks long-term dependencies without loss of information.
5. Rank Aggregation: group the individual results of multiple methods to create a precise ranking of quotes.

Generation of Humour

Even though humour is not the main target in the previous works, some of the previous results may produce a hu-

[2] `https://www.gutenberg.org/`

[3] `http://icwsm.cs.umbc.edu/data/icwsm2009/`

mouristic 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.

As the majority of systems of this kind, all the previous are for English, but we aim to work with Portuguese. In this 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 poems; (iii) an assignment method(s).

The first step was to gather a good collection of data on the chosen context, accomplished by gathering news from the News API[4] sources, found in Portuguese newspapers' online editions[5]. The News API allows a client to get a maximum of one hundred current news on given keywords, e.g. *'clima'* (climate) or *'ambiente'* (environment), returning an object with news whose titles are similar to the keyword. 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'), that were posted on the three months previous to the call of the API (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[6]. 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
- Word Embeddings (WE) generated by GloVe
- Word Embeddings generated by GloVe plus the weighing of the TF-IDF algorithm
- Word Embeddings generated by FastText (FT)
- Word Embeddings generated by FastText plus the weighing of the TF-IDF algorithm
- Bidirectional Encoder Representations from Transformers (BERT)

For GloVe and FastText[7] (using CBOW), we used models pre-trained for Portuguese, with 300-dimension vectors, with GloVe being available as NILC embeddings (Hartmann et al., 2017). For BERT, we also used a pre-trained multilingual model available by Google, covering 104 languages, including Portuguese: BERT-Base, Multilingual Cased[8]

For most of the methods, except BERT, both headlines and proverb were first pre-processed with the NLPyPort package (Ferreira, Gonçalo Oliveira, and Rodrigues, 2019), a layer on top of NLTK tackling Portuguese, specifically. This enabled the linguistic pre-processing, namely with tokenization and PoS tagging, and was essential for further application of the similarity methods.

Afterwards, the similarity of each headline with each proverb is computed and the proverb with the most 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: these vectors were the result of the average of the vectors of the sentence's tokens. Following the computation of the similarity, the proverb with the best 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 the results of each method is a subjective task. Therefore, we had to resort to human opinions, more precisely twenty-four volunteers who were asked to answer a survey. They were grouped into six teams of four people, each assigned to ten headlines, such that each result of the respective similarity method had a total of 30 assessments by different people, summing a total of 240 different evaluations.

In the survey, created on Google Forms[9], for each of the ten assigned news headlines, the person answering had to classify the selected proverbs regarding their relation with the news headline and how funny they were considering this relation.

As an example, for the headline *"Emissões atmosféricas aumentaram em 2017"* ("Atmospheric emissions raised

[4] https://newsapi.org/

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

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

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

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

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

Method	Mean	Median	Mode
Jaccard	2.4 ± 1.15	2	1
Count Vectorizer	2.2 ± 1.06	2	1
TfIdf Vectorizer	2.3 ± 1.09	2	1
WE + GloVe	2.2 ± 1.03	2	1
WE + GloVe + TfIdf	2.1 ± 1.05	2	1
WE + FT	2 ± 1.03	2	1
WE + FT + TfIdf	2.1 ± 1.03	2	1
BERT	2.1 ± 1.04	2	1

Table 1: Representation of how related proverbs were to the news titles.

Method	Mean	Median	Mode
Jaccard	2.5 ± 1.06	2	2
Count Vectorizer	2.3 ± 1.04	2	2
TfIdf Vectorizer	2.3 ± 1.01	2	2
WE + GloVe	2.2 ± 1.019	2	3
WE + GloVe + TfIdf	2.2 ± 0.99	2	2
WE + FT	2 ± 0.97	2	1
WE + FT + TfIdf	2.2 ± 1.05	2	1
BERT	2.2 ± 1.11	2	1

Table 2: Representation of how funny proverbs were when related to the news title.

Method	N^{o} Sel.	Mean	Median
Jaccard	7	2.6 ± 1.27	3
Count Vectorizer	3	3.7 ± 0.58	4
TfIdf Vectorizer	6	2.3 ± 1.03	2
WE + GloVe	2	3.5 ± 2.12	3.5
WE + GloVe + TfIdf	1	2	2
WE + FT	2	2.5 ± 0.7	2.5
WE + FT + TfIdf	1	3	3
BERT	1	3	3

Table 3: Representation of the number of selections (N^{o} Sel.) with more than 3.5 average score and the intersection of tokens between the proverb and the news title.

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 four 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 four point scale: Not funny (1); Remotely funny (2); Considerably funny (3); Extremely funny (4).

In Table 1 and Table 2, we reveal the results of this evaluation, respectively for the relation between proverb and headline and its funniness. From this data, it was possible to compute certain metrics, like the mean, median and mode, to better understand the scope of these results.

Table 3 presents a global evaluation considering only the best proverb selections, which we considered to be those with at least a 3.5 average score on the relatedness ranking.

As previously stated, the proverbs selected by each method were evaluated regarding their relatedness with the headline and whether it had humoristic value or not. Considering Table 1, the majority of the results were not very satisfactory, as most of the proverbs had little relation to their headlines, as seen in the mode of each method. However, in regard to the funniness of the proverbs, on Table 2, the mode varied from method to method, producing some good results, which can mean that the idea of mixing proverbs with news headlines may give the headline a humorous touch.

Discussion

From Table 1 and Table 2, it is possible to declare that theoretically simpler methods achieved the best scores both in the quality of the relation between the selected proverb and the given headline, particularly the simplest one, the Jaccard similarity.

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 relation between them. The same can be said for the second and third methods, the Count Vectorizer and the TF-IDF Vectorizer, which also achieved the next best two scores.

A curious fact is that the more recent methods, namely those using word embeddings, both with GloVe and FastText, and BERT, achieved lower scores. Even though they selected proverbs whose meaning may not be too different from their title's meaning, their relation is, perhaps, not as pinpointed or clear as in the simpler methods. However, despite having lower scores, BERT was able to make a selection which scored an average of 4 points, as seen in the third headline in Table 4, where the chosen proverb's meaning and urgency clearly applies and is related to the headline.

Another one of the best examples of a successful proverb selection regarding the relation between title and proverb, is the first headline in Table 4, which scored 4 for all its 4 testers. Using the TF-IDF Vectorizer method, the system selected a proverb whose meaning may correlate with the title's meaning, as they share the word *”lixo”* (”trash”), for example. On the other hand, one of the worst scores (mean = 1) was also for a proverb selected for this headline, but obtained with GloVe word embeddings, which chose the proverb *”A ordem dos fatores não altera o produto.”* (”The order of factors does not change the end product”), alluring to the commutative property of the multiplication of two real numbers.

With concern to the best funniness related results, as seen in the first headline of Table 5, 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). On the other hand, the worst score regarding the funniness is attributed to the proverb *”Casa de pais, escola de filhos.”* (”Home of parents, school for sons”), in regard to the news title *”Óculos*

ID	News Headline + Proverb	Method
1	*”Malásia devolve 150 contentores ilegais de lixo a países subdesenvolvidos”* (”Malaysia returns 150 illegal trash containers to underdeveloped countries”) + *”Quem faz de si lixo, pisam-no as galinhas”* (”Whoever makes trash out of you, the chickens will stomp them”)	TfIdf-Vectorizer
2	*”Tempestade ’Glória’ fez 12 mortos em Espanha. Governo culpa alterações climáticas”* (”’Gloria’ storm made 12 casualities in Spain. Government blames climate change”) + *”A culpa morre solteira.”* (”Guilt does not die single”)	TfIdf-Vectorizer
3	*“Ainda não é demasiado tarde para salvarmos os oceanos”* (”It is not too late to save the oceans”) + *”Não deixe para amanhã o que você pode fazer hoje.”* (”Do not leave for tomorrow what you can do today”)	BERT

Table 4: Representation of the selections whose ranking of relatedness between headline and proverb scored 4.

ID	News Headline + Proverb	Method
1	*”Veredicto abre a porta a protecção para ’refugiados climáticos’”* (”Veredict opens door for protection to ’climate refugees’”) + *”Para trás mija a burra.”* (”The donkey urinates backwards”)	Jaccard
2	*”Judoca Jorge Fonseca galardoado com o prémio Ética no Desporto de 2019”* (”Judoka Jorge Fonseca is awarded with the prize ’Sport Ethics 2019’”) + *”Não contes com o ovo no cu da galinha.”* (”Do not count on the egg being in the chicken’s butt.”)	Jaccard

Table 5: Representation of the selections whose ranking of funniness between headline and proverb scored 4.

de natação com realidade aumentada” (”Swimming glasses with augmented reality”). Selected by the word embedding plus FastText, it is difficult to grasp the scope of similarity between these two sentences, so much as to find their relation funny.

Table 3 is an indication of the methods with higher global success in terms of 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 the lead is the simplest algorithm, the Jaccard similarity, followed closely by TF-IDF Vectorizer and Count Vectorizer. In regard to the amount of words shared between these chosen proverbs and the news title they are related to, the median of this statistic is higher for the Count Vectorizer and Jaccard. Furthermore, the more complex and recent approaches did not have as many successful selections. However, the chosen selections of these approaches coincided with proverbs with at least two common words with the headline. Therefore, it may be possible to argue that people are quicker to relate two sentences that share the most words, in regards to their full semantic value chosen by the approaches that are more up to date, according to this data.

Conclusions

This study was developed aiming to achieve significant results in the scope of automatic text suggestion, by classifying different textual processing approaches in regard to the Portuguese language and culture. Using news headlines as input and a Portuguese proverbs corpus as texts for selection, we developed a system able to apply different methods for the task of selection the most related proverbs to each headline. An application of this kind could possibly be useful for journalists to a certain degree, as it suggests proverbs they could use to make their news more appealing.

The obtained results, given some thought, 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 which were selected by state-of-the-art methods, while having a supposedly better relation to the headline due to their context similarity, did not score as many points amongst our volunteers.

In terms of future endeavours, a possible option to research would be to test supervised approaches for Semantic Textual Similarity (STS) in this scenario, including, for instance, 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). We may also test the same approach with the recently available BERT model trained for Portuguese (Souza, Nogueira, and Lotufo, 2019), or try training a model with Twitter 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 tweet. We should also stress that, even though our experiments were limited to proverbs and sayings, it should also work with other well-known expressions, like song titles or movie titles, which is something we might tackle in the future.

Acknowledgments

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

References

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 Creativity (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.

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. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

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.

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.

Jurafsky, D., and Martin, J. H. 2019. *Speech and Language Processing (3rd Edition)*. Draft chapters available from `https://web.stanford.edu/~jurafsky/slp3/`.

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.

This page is intentionally left blank.

Appendix B

This appendix presents the short paper submitted in International Conference on Computational Creativity (ICCC) 2020.

Exploring Word Embeddings for Text Adaptation to a given Context

Abstract

We propose three methods for the adaptation of well-known sayings (e.g., proverbs, movie titles) for a context, given by a textual input (e.g. news headline). These methods – word substitution, analogy and vector difference – present different ways of exploiting word embeddings for word replacement towards a new text that should be semantically related to the input, in the sense that it could be used as a sub-title or comment, though more creative. They were further combined with two selection methods, based on word overlap and on sentence embeddings, and used in the production of text in context with a small set of Portuguese headlines. To better understand how well our purpose was suit, results were manually assessed. All methods produced text with both syntax and relatedness to the input above average, contrasting the underachieving funniness scores.

Introduction

To amplify the range of a given story, either real or made-up, authors commonly reuse expressions or sayings known by a general audience as a title or subtitle, sometimes also achieving a humorous effect. If the saying is related enough, it can be used directly, but it may also suffer minor adaptations, to become more related to the context and still resemble the original saying. Working on the automation of this process is thus natural. In fact, in scope of linguistic computational creativity, related systems have been developed for generating new creative headlines by resorting to figurative language (Alnajjar, Leppänen, and Toivonen 2019), or blending them with well-known expressions (Gatti et al. 2015); poetry inspired by news stories (Colton, Goodwin, and Veale 2012; Chrismartin and Manurung 2015) or Twitter trends (Gonçalo Oliveira 2017); or applying metaphors to the current news (Veale, Chen, and Li 2017). Having in mind the transformation of text with replacements constrained by the given intentions, operators on word embeddings were even formalised (Bay, Bodily, and Ventura 2017). Other systems simply recommend quotes to be used in dialogues (Ahn et al. 2016), or assign proverbs to news headlines (Anonymous 2020).

As a complement to both previous works, and following the idea of using word embeddings, we propose three different methods that exploit this kind of word representation for adapting selected text, so that relatedness to the context increases: substitution of a word by one from or related to the context; substitution of two words by two analogously-related, one of which is from the context; and substitution of two words related to the context, in such a way that their relation is preserved.

Although the proposed methods are language dependent, this study is focused on Portuguese, so we rely on Portuguese sayings, namely proverbs and movie titles. Simple selection methods, based on TF-IDF and on sentence embeddings (BERT encodings), were used for selecting an initial set of sayings to be adapted and for making the final selection of the result to exhibit, out of all of those produced.

To better understand their potential, we ran different combinations of selection and adaptation methods and assessed their results for 30 headlines, manually. For all methods, syntax was generally good, relatedness was above average and funniness below. Simple selections of the most similar saying out of the original list, by the selection methods, were assessed with the same criteria, with TF-IDF having comparable scores and BERT clearly lower. This suggests that the proposed adaptation methods are capable of creating new text, from a lower amount of original examples, and of comparable quality.

The paper is organised as follows: after this introduction, the proposed methods are described; the results of the manual assessment are then presented and discussed; finally, we conclude with a brief discussion.

Methodology

Our main goal is to adapt a known saying so that it becomes as related as possible to a given short text, in such a way that it can be used as a more creative way of transmitting the same idea, complementing it or just commenting on it.

We propose three automatic methods for this: Substitution, Analogy and Vector Difference. Besides a list of well-known sayings, to be modified based on the input text (e.g., news headline), all methods: (i) exploit a pre-trained model of static word embeddings, where words are represented by dense numeric vectors; (ii) assume that the most relevant words in a text are previously computed, which can be done with methods like TF-IDF, or simply by considering the open-class words (nouns, verbs, adjectives) that are used in a large corpus but have the lowest frequency; (iii) go through

Method	Headline	Proverb	Output
Substit	*Bancos preparam-se para dar menos crédito às famílias* (Banks preparing to give less credit to families)	*amigos, amigos, negócios à parte* (Friends, friends, business apart)	*bancos, bancos, negócios à parte* (Banks, banks, business apart)
Analogy	*EUA estão a apontar para o pior número de desemprego da sua história* (USA are pointing out to the worst unemployment numbers in their history)	*não deixes para amanhã o que podes fazer hoje* (Do not leave for tomorrow what can be done today)	*não comeces para amanhã o que podes apontar hoje* (Do not start tomorrow what you can point out today)
VecDiff	*Finge ter Covid-19 no Facebook e acaba detido* (Pretends to have Covid-19 on Facebook and ends up arrested)	*quem com ferro fere, com ferro será ferido* (Those who hurt with iron, with iron shall be hurt)	*quem com ferro finge, com ferro será detido* (Those that pretend with iron, with iron shall be arrested)

Table 1: Running examples of the application of each adaptation method.

all the sayings in a list and try to make adaptations focused on the most relevant words of both the sayings and the input texts. Methods only differ on the adopted strategies for selecting the word(s) to replace.

The first method, Substitution, replaces the most relevant word in the saying, a, by a word from the input text, b. Our intuition is that, by using a relevant word of the input text, the meaning of the saying becomes more semantically-related to the given context.

The second method, Analogy, relies on a common operation for computing analogies in word embeddings, i.e., $b - a + a^* = b^*$ (Mikolov, Yih, and Zweig 2013), phrased as b^* *is to* b *as* a^* *is to* a. The strategy is to use the two most relevant words in the saying as a and a^*, and the most relevant word in the input as b. Then: (i) from the previous three, compute a new word b^*; (ii) in the original saying, replace a and a^*, respectively by b and b^*. Given that both pairs of words are analogously-related, our intuition is that the result will still make sense and be more related to the input text.

The third method, Vector Difference, also selects the two most relevant words in the input text, b and b^*, and then: (i) computes the vector between the previous $b{-}b^*$; (ii) identifies the pair of open-class words in the saying, a and a^*, such that $a{-}a^*$ maximises the (cosine) similarity with $b{-}b^*$; (iii) replace a and a^* respectively by b and b^*. Our intuition is that the new text will not only use two words of the input, and thus be more related, but also that they will be included in such a way that their relation is roughly preserved.

Table 1 shows an example of the application of each method, including the original headline, a proverb and the resulting output. Replaced word and their replacements are underlined. In the first example, $b = bancos$ replaces $a = amigos$. In the second, $a = deixes$, $a^* = fazer$ and $b = apontar$, with $b* = b - a + a^* = comeces$. In the final example, $a = fere$, $a^* = ferido$, to which, out of the words in the headline, $b = finge$ and $b^* = detido$ is the pair with the most similar difference.

To avoid syntactic inconsistencies, for any method, replacement candidates must match the morphology of the replaced word, including part-of-speech (PoS), gender and number, obtained from a morphology lexicon and / or by the application of a PoS tagger. If morphology does not match, the lexicon can also be used to inflect the candidate to the target form. Only if it is not possible, the saying is not considered. In any case, the set of possible replacements can be augmented by considering not only the relevant words in the input text, but also the most semantically-similar words, computed in the embeddings. For example, in the Substitution method, a can be replaced by a word different but semantically similar to b.

Finally, running through all available sayings should result in several new texts. Even if, due to the morphology constraints, some sayings end up not being used, others will. Also, if similar words are considered for the same input, the same method may produce several variations of the same text. Therefore, the final step selects the text to be used. Out of the produced texts, this will be the most similar with the input, according to a sentence similarity method (e.g., TF-IDF, BERT encodings).

We should add that, to avoid going through the full list of available sayings for each input text, an initial selection of sayings to use may also be done. The choice may rely on one of the aforementioned similarity methods and select the n most similar original sayings, as well as m randomly chosen, to improve diversity.

Results and Evaluation

To take some initial conclusions, we ran all the methods in a set of 30 news headlines. Then, the results of each method were manually assessed by two human judges. Besides some insights on the suitability of each method for our purpose, we tested combinations of two selection methods – TF-IDF and BERT – and included in the results also original sayings directly selected by each of them. Both selection methods get the most similar sayings to the input context. The difference is that TF-IDF represents each text as a weighted vector, based on all the sayings, while BERT encodes each text as a 768-sized vector according to pre-trained model covering 104 languages[1]. Selection methods were used in two different stages, initial selection of sayings to use and final selection from the adapted sayings, meaning that 14 texts were obtained for each headline: four produced by each method with a different combination of selection methods, plus two by each selection method alone. Adaptation methods were applied to a set of 100 sayings, including the top-50 related according to the selection method ($n = 50$) and a random selection of other 50 sayings ($m = 50$).

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

Method	Syntax			Relatedness			Funniness		
	Mo	Means±StDev	Md	Mo	Means±StDev	Md	Mo	Means±StDev	Md
Substitution	3	2.94 ± 0.24	3	3	2.06 ± 0.88	2	2	1.71 ± 0.68	2
Analogy	3	2.91 ± 0.28	3	3	2.31 ± 0.76	2.5	1	1.64 ± 0.66	2
Vector Diff	3	2.87 ± 0.33	3	3	2.51 ± 0.72	3	2	1.81 ± 0.64	2
TF-IDF	3	3.00 ± 0.00	3	3	2.35 ± 0.83	3	1	1.62 ± 0.66	2
BERT	3	3.00 ± 0.00	3	1	1.70 ± 0.90	1	1	1.47 ± 0.64	1

Table 2: Evaluation figures.

Although the proposed methods are language dependent, we focused on Portuguese, our mother tongue. Our list of sayings included 1,600 Portuguese proverbs from project Natura[2] and over 3,000 movie titles in Portuguese, from IMDB[3]. Most should be well-known, with proverbs being part of the quotidian of most Portuguese people, where they are used to emphasize certain situations, usually implying some kind of humour. Moreover, these sayings are not usually to be taken literally, as they use several stylistic variations and their underlying meaning may not be straightforwardly understood by a computer.

For the processing of Portuguese, we used LABEL-Lex (Ranchhod, Mota, and Baptista 1999) as the morphology lexicon, NLPyPort (Ferreira, Gonçalo Oliveira, and Rodrigues 2019) for PoS tagging and lemmatization, and a pre-trained GloVe model with 300-sized vectors (Hartmann et al. 2017). The selection of the most relevant words relied on their frequency in the newspaper corpus CETEMPúblico (Rocha and Santos 2000).

For each headline, judges were presented with the headline followed by the list of texts by each combination and were asked to use a 3-point Likert scale for ranking: syntax (1, the expression has several grammatical and/or structural issues, and may be difficult to interpret; 2, the expression has minor issues regarding grammar and structure, but is still understandable; 3, the expression does not have any grammatical or structural issues); relatedness (1, minimal or no relation at all between the generated expression and the input; 2, somewhat related to the input; 3, relation with the headline is clear / could be used as a substitute or a comment); and funniness (1, not funny and will not make anyone laugh; 2, somewhat funny and could be potentially be funny, depending on the reader's subjective view; 3, very funny, with a great potential to make people laugh). Table 2 shows several figures of this preliminary evaluation, namely the mode (Mo), means and standard deviation, and median (Md) of the scores for texts produced by each adaptation method, regardless of the judge and selection methods, plus the results of the two selection methods when applied directly to the full list of sayings.

Judge agreement was measured with Cohen's Kappa, resulting in 0.57 (moderate) for syntax, 0.35 (fair) for relatedness and 0.17 for funniness (fair). Syntax is more objective, and thus agreement was higher. On the other hand, the other two aspects, especially funniness, are highly subjective, also due to the structure and figurative language of the Portuguese proverbs and vagueness of some movie titles.

According to the scores, the syntax is not severely affected by the adaptations, meaning that the produced text is generally grammatical. Few exceptions occur in the adaptation of verbs. Specifically, in Portuguese, the same verb has often different forms for different tenses, genders and numbers, but the same form may also work for different tenses. Thus, incorrectly identifying the tense in the original saying may result in using an incorrect form in the adapted text.

Regarding relatedness, results are above average, i.e. means higher than 2 and mode equals 3. Despite a slight trend of Vector Difference getting higher scores and Substitution lower, especially looking at the median, the standard deviation also suggests that this difference is not significant enough. Though, differences would make sense because, while Substitution makes a single replacement, the other two methods replace two words that, nevertheless, try to keep original relation, with the Vector Difference generally using two words from the original context.

On funniness, results were not as good as for the other aspects. This may be due to the aforementioned subjectivity of humour, which may hamper the judge's decision to give the maximum funniness to a text, because actually making other people laugh depends on many variables. Moreover, the capability of producing content with humoristic value is highly dependent on the context of the input, e.g. it is harder to generate content regarding sad news headlines.

Table 3 illustrates some of the results produced. The first three got the maximum score in all aspects by all judges, and the final two got the lowest scores in relatedness and funniness. Furthermore, it is important to state that most of the generated expressions that scored the minimum in both relatedness and funniness used BERT for their final selection, as it seems to suffer from the figurative language used in the sayings, and often selects one that is too distant from the headline with less focus on shared words. On the other hand, TF-IDF tends to select expressions that share words with the input, thus increasing their relatedness and achieving scores similar to the adaptation methods. This should, however, be analysed more deeply in future work.

Conclusion

Briefly, this study proposed three text adaptation methods to bring a well-known saying closer to a given context, with positive results regarding syntax and relatedness to the context, but not so much on funniness. When compared to the usage of existing sayings, selected with TF-IDF, with no adaptation, scores are very similar. This shows that

[2] https://natura.di.uminho.pt

[3] https://www.imdb.com/interfaces/

Method	Headline	Proverb	Output
Substit + TF-IDF	*Rooney sobre cortes de salários: 'Porque é que são os futebolistas os bodes expiatórios?'* (Rooney about salary cuts: 'Why are footballers the scapegoats?)	*Os amigos são para as ocasiões* (Friends are for the ocasions)	*Os expiatórios são para as ocasiões.* (Scapegoats are for the ocasions)
Analogy + TF-IDF	*Bancos dizem que as condições das linhas de crédito foram definidas pelo governo* (Banks claim that credit conditions were defined by the government)	*paga o justo pelo pecador* (The fair pays for the sinner)	*paga o definido pelo pecador* (The defined pays for the sinner)
VecDiff + BERT	*Ronaldo juntou a família na quarentena para cantar os parabéns à sobrinha* (Ronaldo brings family together during quarentine to sing happy birthday to his niece)	*papagaio come o milho, periquito leva a fama.* (Parrot eats the corn, but the parakeet gets the fame)	*papagaio come o milho, sobrinhinho leva a fama.* (Parrot eats the corn, but the little nephew gets the fame)
Substit + BERT	*Finge ter Covid-19 no Facebook e acaba detido* (Pretends to have Covid-19 on Facebook and ends up arrested)	*Nem por ser Natal* (Not even for being Christmas)	*nem por ter natal* (Not even for having Christmas)
Substit + BERT	*Trabalhadores da hotelaria e turismo há quase dois meses sem salários* (Workers from hotels and tourism have been without salary for two months)	*Mãe só há uma* (There is only one mother)	*Semana só há uma* (There is only one week)

Table 3: Examples of produced texts, along with their adaptation and selection methods.

the adaptation methods are indeed capable of creating new syntactically-correct and related text, and thus a good option when the list of sayings is limited. We recall that the selection methods were applied to the full list of 4,600 sayings, while adaptation methods used only a selection of 100 sayings, which they were able to adapt for increased relatedness.

For future endeavours, it would be prolific to test and assess social reactions to the produced text, through the inclusion of the system on a Twitter bot, a chatbot (as a possible conversational aid), or even just as a creativity booster, to be used in certain areas, e.g. journalism. A wider evaluation, involving more people, is also planned for helping us pinpoint future improvements, as well as determining other possible applications for these methods.

References

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 *Procs of 10th ICCC*, 258–265. Association for Computational Creativity.

Anonymous. 2020. Omitted for review purposes. In *Omitted for Review purposes*, Accepted for publication.

Bay, B.; Bodily, P.; and Ventura, D. 2017. Text transformation via constraints and word embedding. In *Proc. 8th International Conference on Computational Creativity*, ICCC 2017, 49–56.

Chrismartin, B., and Manurung, R. 2015. A chart generation system for topical meaningful metrical poetry. In *Procs. of 6th ICCC*, ICCC 2015, 308–314.

Colton, S.; Goodwin, J.; and Veale, T. 2012. Full FACE poetry generation. In *Procs. of 3rd ICCC*, 95–102.

Ferreira, J.; Gonçalo Oliveira, H.; and Rodrigues, R. 2019. Improving NLTK for processing Portuguese. In *Procs. SLATE 2019*, volume 74 of *OASIcs*, 18:1–18:9. Schloss Dagstuhl.

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

Gonçalo Oliveira, H. 2017. O Poeta Artificial 2.0: Increasing meaningfulness in a poetry generation Twitter bot. In *Procs. of the Workshop on 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).*

Mikolov, T.; Yih, W.-t.; and Zweig, G. 2013. Linguistic regularities in continuous space word representations. In *Procs of 2013 Conference of the North American Chapter of the ACL: Human Language Technologies*, 746–751. ACL.

Ranchhod, E.; Mota, C.; and Baptista, J. 1999. A computational lexicon of Portuguese for automatic text parsing. In *Procs of SIGLEX99 Workshop: Standardizing Lexical Resources*. ACL Press.

Rocha, P. A., and Santos, D. 2000. CETEMPúblico: Um corpus de grandes dimensões de linguagem jornalística portuguesa. In *V Encontro para o processamento computacional da língua portuguesa escrita e falada (PROPOR 2000)*, 131–140. São Paulo: ICMC/USP.

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