FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Computing the accuracy of an automatic system for relevance detection in social networks

Filipe Fernandes Miranda

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Álvaro Figueira

July 23, 2017

Computing the accuracy of an automatic system for relevance detection in social networks

Filipe Fernandes Miranda

Mestrado Integrado em Engenharia Informática e Computação

Approved in oral examination by the committee:

Chair: Professor Carla Teixeira Lopes

External Examiner: Professor Ricardo Campos

Supervisor: Doctor Álvaro Figueira

July 23, 2017

Abstract

To correctly assert the precision of a classification model, previously labeled data is needed to validate the output provided by the model. The process of labeling data can be achieved either by a human manual effort or, automatically, by computers.This dissertation reports the creation process and development options of an automated system to assess the precision of a classification model where no human component is used throughout the process of labeling the data.

The goal of the classification model, used as the basis of this project, is to identify newsworthy social network messages. The model takes advantage of the vast information spread across social networks and aims to filter relevant data, which may have important information from a journalistic point of view.

To assert the precision of the classification model, social network messages need to labeled as news-worthy or not, which can be achieved by paid manual labeling. While this assessment is fundamental to train the model at a first stage, the monetary, time and precision costs involved do not allow this procedure to be done regularly. Yet, the classification of data is essential to train our models and to determine their accuracy.

For this reason, and to avoid the downsides of manual labeling, a four stage automatic system was created. This new approach starts with the collection of data, both messages and news articles. The collected messages will be classified based on the news articles also gathered.

The second step is the information extraction. Here, the system will analyze the information present in the different texts,using several information extraction techniques, such as named entity recognition and keywords detection. This results in a standardized vector of features for the messages and news.

The third stage is the matching of news and social media messages, based on the similarity of contents. When a message is associated with the content of a news article, it is labeled as news related. This final part, message classification, allows the distinction of news relevant and not relevant messages. This process is also helped by a filtering model, which helps exclude weak matches. These are cases where even though messages and news have similar information, it is not relevant or newsworthy.

The matching system was validated while it was being developed. The final system has a precision of over 80% in labeling newsworthy social network messages.

Nonetheless, techniques and mechanisms developed in this thesis can be extrapolated for other uses within the media and journalism world. As an example, the research can be targeted at finding possible contradictory information in social network messages, potentially helping news entities to update their stories as live information comes through. Another application might be to detect breaking news and crisis events.

Resumo

De forma a calcular a precisão de um modelo de classificação, são necessários dados já categorizados para validar os resultados obtidos pelo modelo. O processo de categorização pode ser feito manualmente ou de forma automática. Nesta dissertação, é apresentada a implementação de um sistema automático, capaz de aferir a precisão de um modelo de classificação sem qualquer interferência humana na categorização de dados. O presente modelo de classificação pretende identificar mensagens de redes sociais, que sejam relevantes no contexto jornalístico. O modelo faz proveito da imensidão de dados fornecidos por diversas redes sociais de forma a filtrar mensagens com informação potencialmente relevante para jornalistas ou organizações noticiosas.

Para afirmar a precisão deste modelo, mensagens e publicações de redes sociais terão de ser categorizadas como sendo relevantes ou não, o que pode ser feito através de um trabalho manual remunerado. Embora esta categorização seja fundamental para treinar o modelo numa primeira fase, acaba por se tornar inviável para o projeto executar esta constante categorização devido aos custos de precisão, monetários e de tempo que acarta. Contudo, a categorização de mensagens e publicações continua a ser necessária para calcular a precisão do modelo.

O objetivo desta dissertação é então desenvolver e implementar um sistema automático capaz de avaliar regularmente a precisão do modelo de classificação, ou seja, a precisão com que o modelo identifica corretamente mensagens com informação potencialmente relevante. Desta forma, o sistema deverá ser capaz de automaticamente categorizar mensagens e publicações de redes sociais quanto à sua relevância jornalística.

Para uma avaliação sistemática do modelo de classificação, foi desenvolvido um agregador de mensagens e artigos quer de redes sociais quer de organizações noticiosas. Posteriormente, o sistema utiliza os dados recolhidos para averiguar a informação comum a mensagens de redes sociais e artigos noticiosos. Se uma mensagem apresenta um conteúdo semelhante ao de um artigo, então deduz-se que tenha algum grau de relevância noticiosa, e será automaticamente classificado com relevância noticiosa.

Um sistema de correspondência foi desenvolvido para associar cada mensagem de rede social ao artigo de notícias mais relevante, presente na base de dados do sistema. Para detetar casos onde uma mensagem e artigo de notícia têm conteúdos idênticos ou semelhantes, técnicas de extração de informação, como reconhecimento de entidades e deteção de palavras-chave, foram aplicadas aos dados, resultando num conjunto de features para cada um dos textos.

Um modelo de filtragem foi também desenvolvido para excluir correspondências fracas. Ou seja, casos em que as informações comuns a mensagens e notícias não são relevantes no contexto jornalístico. Caso contrário, as mensagens correspondidas são classificadas como relevantes.

Ao longo do seu desenvolvimento, o sistema de correspondência foi regularmente validado. O sistema final é capaz de classificar as mensagens nas redes sociais como relevantes, com uma precisão de mais de 80%.

No entanto, as técnicas e mecanismos desenvolvidos nesta dissertação podem ser extrapolados para outros usos no mundo jornalístico. Por exemplo, na deteção de informação potencialmente

contraditória em redes sociais, o que pode ajudar entidades noticiosas a atualizar as suas histórias, enquanto recebem novas informações. Outra utilização é a deteção de noticias de última hora ou de eventos de crise.

Acknowledgements

I would firstly like to thank my thesis advisor Álvaro Figueira, for the guidance, support and prompt availability to answer all my questions and help me throughout the development of this research.

I would also like to express my gratitude to FCT (Portuguese Foundation for Science and Technology), which gave me the opportunity of a research grant supported by the ERDF – European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and also by National Funds through the project "Reminds/UTAP-ICDT/EEI-CTP/0022/2014".

To all the teachers, staff and colleagues at the University of Porto, who have guided me in this incredible journey of learning and progress.

To my friends and lab partners Nuno and Filipe, for the continuous help and patience. To Jorge and Catarina for never letting me drown in work and reminding me that there is a world outside the computer.

To my parents and sister, I thank you from the bottom of my heart for all the uplifting motivation you provided me all these years and for always believing in me.

Filipe Miranda

"Truth is ever to be found in simplicity,
and not in the multiplicity and confusion of things"

Isaac Newton

Contents

List of Figures

LIST OF FIGURES

List of Tables

Abbreviations

AUC	Area under the curve
FB	Facebook
FN	False negative
FP	False positive
IE	Information Extraction
POS	Part-of-speech
RSS	Rich Site Summary
SNN	Social Network News dataset
SNRSS	Social Network RSS feed dataset
SNT	Social Network Trending topics datset
SVM	Support vector machine
TN	True negative
TP	True positive
TW	Twitter
URL	Uniform Resource Locator

Chapter 1

Introduction

In this first chapter, the context and motivation behind the project will be explored, as well as some key ideas and approaches to the problem of computing automatically the accuracy of a relevance detection system in social network texts.

1.1 Context

The central role that social networks have taken in the lives of millions of people, changing the way they communicate and share information, has provided with an immensity of untapped and studied data that can prove to be valuable. Due to the growth of these platforms, information extraction from social networks has been a recurrent topic of research in recent years.

Social networks allowed for anyone who has access to these platforms to have a voice and means to share information regarding subjects or events happening around them, sometimes even before being reported by news-wire [PO13, CMP11]. Users were now allowed to be the reporter on the scene, sharing and commenting on incidents and affairs. Two of the main social networks, Facebook and Twitter, have become a news information channel, due to their continuous flow of information in real time and ease of broadcast of information [MP15, LCLZ].

Although the opportunity for the citizen journalist has appeared in recent years in social networks, news organization have found the channel to be useful for their own coverage. Even though the public will usually still follow stories and events through professional media outlets [Mur11], journalists embrace and use information on social networks to contact and gather details to better inform their publics.

Extracting and detecting newsworthy posts from the vast information spread across all social networks is of extreme value to news agencies and other organisms to gather information, but also a difficult task due to do the vastness of messages published and shared on these same social networks.

In order to develop such a system, able to filter out irrelevant information, the system needs to have access to labeled data (posts from social networks). Labeled data will enable researchers to evaluate the accuracy of the system being developed, that is if the system is correctly identifying relevant and irrelevant information, and guide the research in improving that system.

The labeling of data can be achieved either by manual effort with human classifiers or automatically using computers to label the data.

A labeling approach performed by humans can have a high cost, since people should be remunerated for their effort (monetary cost) and is not an automatic labeling (time cost). Considering the time it takes to label the data and a restrict budget, the labeled dataset could quickly become outdated. A tight control over who labels the social network posts is also important to decrease human error and avoid random classification.

The development of an automatic system capable of doing equivalent work in labeling the data of social networks would allow researchers to assess the accuracy of systems being developed, with no added costs and up-to-date performance - fitting for a sustainable development.

1.2 Project

This thesis was developed as part of the research project "Relevance Mining Detection System" (REMINDS/ UTAP-ICDT/EEI-CTP/0022/2014), which aims to develop a classification model capable of detecting relevant information, from a journalistic point of view, in the vast pool of posts, tweets and comments in social networks such as Facebook and Twitter.

The drive behind this research is to take advantage and improve the access to relevant information shared in social networks. The “filter” developed in this project could provide journalists and news organizations with a powerful tool to improve their coverage of news events or even detect incoming relevant information in real time.

1.3 Motivation and Goals

Research projects working with social network data, such as REMINDS, usually rely on manual effort to label the data needed to develop, train and evaluate their systems.

The classification of data by humans can prove to be an obstacle in the growth and development of any project. Since this process is not automatic, the outcome of the manual classification can be influenced by many factors:

1. The definition of the labeling classes and its specificity: labels should be defined with clarity and leaving no room for different interpretations so that multiple opinions can be avoided in the classification process.

2. The knowledge of the classifiers: the perception of newsworthiness relies heavily on the knowledge that a human classifier has of the news. While the classifiers might be labeling

the messages following the definition provided, their knowledge of the topic in question might be scarce, which can strongly influence the correct or incorrect labeling of data.

3. The number and reliability of classifiers labeling the data: while fewer classifiers might seem to be the best option, as we cloud certify that these understood the classifications, the wrongful labeling by one person could compromise the entire classification. On the other hand, if the classification is performed by several classifiers, their agreement should be studied to remove untrustworthy classifications.

In addition to the precision uncertainty of manual classification and the time it takes to label the data manually, this usually requires a monetary investment to compensate human classifiers.

Therefore, the goal of this thesis is to provide with a viable solution to human classification: automatic classification. The work presented in this dissertation is focused on developing a labeling tool capable of classifying messages on their newsworthiness so that it can be used by REMINDS and other research projects. This will allow the REMINDS system to automatically assess its precision: comparing the classification coming from the classification model with the labeling performed by the developed tool.

In manual labeling, human classifiers based their opinion on the knowledge they have in the news. This means that the classification of messages is inherently related to the presence, or absence of, their content in the news wire. This relation of newsworthiness and presence in the news will be used by the automatic labeling system developed to classify its data as relevant or not: the presence of the content of a social network message in the news will indicate that the message is newsworthy.

The developed labeling tool tries to overcome all the defects than human classification can bring to the research project:

1. A precise, **exact definition** on the labeling class allows the system to emulate the interpretation of newsworthiness by a computer, classifying the data based on that assumption.

2. A **knowledge base** of news was built as an integral part of the automatic classification system. This will provide with a wider and limitless range of topics that the system recognizes and has information about, when comparing with human classifiers

3. An **automatic system**, meaning that the classification of messages can be done in real time. The tool was built to automatically update itself with new information that can be immediately used, improving the labeling system and, as a consequence, the assessment of the classification model's accuracy.

While the goal of the presented work is to improve the development of project REMINDS, the motivation behind this dissertation is to provide with an alternative venue to label data. Allowing a more sustainable and up-to-date system, that could ultimately help in the development and advancement of scientific research.

1.4 Document structure

The dissertation report is composed of five chapters, including this Introduction, and one Appendix section:

- In Chapter 2, a thorough revision of related work and relevant knowledge regarding news aggregating systems is presented.
- In Chapter 3, we explore related projects, relevant information and knowledge regarding information extraction and similarity measures.
- In Chapter 4, the problem specification and proposed solution are formally described, as well as the implemented system.
- The synthesis, discussions and contributions of this dissertation, as well as future work and other applications of the developed research are presented in Chapter 5.

- In the Appendix A, information and data relevant to this dissertation is presented.

Chapter 2

News aggregation

A news aggregation system provides an assemble of news from a variety of sources in one place. Its aggregation could be performed manually or by an automated system that goes through a variety of news entities' websites, extracting the useful information from each page. Another alternative is to rely on RSS (Rich Site Summary) feeds, where news sites and other publishers provide their content in a specific format, on a regular basis.

In this chapter, work related to the collection of news articles text is reviewed. This study will allow for a better and deeper understanding of the state of the art in this area and in which manners it can be addressed in this thesis.

2.1 Web scraping

Websites are built around human interaction, where information is delivered in structured data such as HTML. Although information is easily interpreted by the user interface, relevant text data is delivered in different patterns, depending on the source, which is not suitable for automatic processes.

The central task of web scrapping is to extract and collect the relevant textual data from sources, eliminating clutter such as code related to the user interface or publicity on the website. This task could be described in three sub-tasks [MRMN14]:

- Collect the HTML files from a source or website.
- Determine the patterns behind which the textual information is presented in the files.
- Apply the recognized patterns to extract data in the desired output format.

The main obstacle when developing web scrapping is the absence of rules or protocols when coding and embedding data in the source code of websites. Therefore, web scrapping has its maximum accuracy when site-specific coded[RGSL04].

Since there is no structure or rigid guidelines followed by media outlets and news organization when building their website and source code, there is no exact implicit structure to the data present in these pages. To design the perfect web scrapping technique, one would have to look at all the pages in a website and develop the script for that one site. Although some values might be carried on to other news websites, there is no consistency in its file and code structure.

There are ways to generalize web scraping for news articles. Since most of the news websites rely on presenting data provided by a database, is safe to presume that pages inside news organizations websites will follow a common layout or template. This way, it is possible to systematically identify news articles titles, bodies or dates of publishing, in a certain header or paragraph, for a given news website.

One approach to generalizing web scraping for news extraction was presented by [RGSL04]. The approach is based on the concept that a Web page can be represented by a tree and the similarity between pages can be calculated by measuring the tree edit distance between the two trees (pages). The tree edit distance is given by the minimum number of operations (vertex removal, insertion or replacement), or cost associated, needed to transform one tree into the other. Clustering the pages with high similarity allows for the extraction of patterns and data matching, providing a generalized and automatic data scrapping system.

Besides extracting the textual information present in the source file, it would not be viable to feed the system each link to a web page manually. It is necessary to design and develop a system capable of automatically explore pages and organizations (web crawler), or use a platform that provides information in a structured way (RSS feed).

Newspaper is a Python library that provides web scraping for news articles. This library allows for the extraction of the article's title, text, author, published date and keyword extraction from the text, among others [New17], by feeding the link to the library's function.

2.2 Web crawler

A web crawler is a process that systematically and automatically browses the Internet. It can be used for indexing web pages, search engines or for data aggregation.

The process starts from a given URL and scans all its content, looking for relevant data and links to other pages. The system then stores the relevant information of a website and continues the process to other links found in the content.

As stated by [MJD07], web crawlers can be classified as generic crawlers or focused crawlers. A generic crawler crawls documents of different topics, usually used by search engines to index sites. Focus crawlers are limited to pages or documents referring to certain topics or domains.

A focused web crawler can be used to aggregate news, by exploring sites and restricting the process to news related sites. Using web scrapping, the crawler extracts the relevant information from a web page, storing it, and continues the process to other links referenced in the text.

Is essential to keep track of visited links to avoid infinite loops and restrict the number of hops from the original news website to control the amount of information gathered.

2.2.1 Web crawler evaluation

A focused web crawler can be evaluated based on its ability to retrieve relevant pages to the topic in focus [MPSR01]. Relevant pages might be hard to define, so there are two types of evaluations that can be used.

User-based evaluation of pages retrieved by the crawler could be conducted, but the sheer volume of evaluations and time needed for a meaningful assessment of a news aggregation system would turn out to be impractical.

An automatic system for evaluating the performance of a focused web crawler would classify a retrieved page as relevant if its content is on topic, which improves dramatically the time needed for classification. This type of classification has the downside of not penalizing recurrent content, that is, near-duplicate pages.

Even though user evaluation would be preferable, the advantages of an automatic system surpass its drawbacks.

A state-of-the-art tool for web crawling is Apache Nutch 1.x. This open source web crawler provides with interfaces to explore, parse and index web pages [nut17].

2.3 RSS feeds

Rich Site Summary (RSS) is a family of XML file formats containing summarized information of websites, mainly used by news organizations. This standard allows a news outlet or any other entity to create a feed of information posted on the site, in a structured and standardized form, with metadata relevant to the publications such as date, author and topic.

Aggregation of news articles could be achieved by combining several RSS feeds from different news sources. Since RSS feeds provide information stripped from clutter, web scrapping is not usually needed, simplifying the task of collecting news articles.

Using RSS feeds in combination with a web crawler could be beneficial to the collection of information. RSS feeds could provide with the initial URLs and the crawler explore them, making the collection of articles more complete and not bound to specific news sites.

2.3.1 RSS feeds services

Most of the well-known news organizations, such as the BBC, CNN and The New York Times [BGS06], provide RSS feeds to their news.

Since news organizations partially rely on add-based revenue for its online public, RSS feeds, while still provided, usually lack crucial information such as a well formed summarized text, in order to force users to go to the original website. With the URL to the news article, using a web scrapping technique, the lack of information or content can be overcome.

Bigger news organizations usually provide with topic related RSS feeds, for a curated stream of news regarding a topic (as economy or technology) or location (Europe, Africa or Asia as an example).

While not an RSS feed, the New York Times Developer Network provides a REST API, capable of filtering through and provide a feed of news articles published by The New York Times, based on the publishing date (a timeframe can be specified) or topics [NYT17].

2.4 Metadata from News Articles

Metadata captures the characteristics of data, concrete or logical interpretations of the content [Fre95]. Metadata in news articles can refer to the author's name, publication's date or to the topic of the article, for example.

The idea behind Semantic Web (metadata) is to provide standards to a data-oriented structure detached from the presentation layer of a web site. The knowledge representation provided by the standards allow for the interpretability of information, by machines and automatic systems, that would have been difficult to achieve without it.

Extracting information from a news website page is not only the collection of its text but also all the metadata regarding the publication. This data might not be presented to the user in the form of user interface, but is usually coded in the source code of the page, using a framework, protocol or structure.

2.4.1 Resource Description Framework

An approach to making the web more "computer friendly" is to use a metadata data model such as Resource Description Framework (RDF), that attempts modeling the description of the information in a structured and systematic manner [CK04].

Open Graph is one of the protocols that uses RDF. The Open Graph protocol was created by Facebook and provides a framework to code metadata related to website pages [Hau10].

It is commonly used by news agencies since Facebook uses this protocol to extract information from pages to provide a better presentation in its platform. The Open Graph protocol builds upon existing technologies of RDF and provides with a simple and powerful tool to bind metadata to web pages.

Open Graph relies on tags and values. Each tag corresponds to a property of the page such as title, type of publication or author. Using a simple web scraper on the web page source file, it is possible to retrieve data encoded in the Open Graph format. Some examples of tags available in the Open Graph protocol:

```
1  <meta property="og:title"Title of the article" />
2  <meta property="og:description" content="Summary of the article" />
```

```
<meta property="og:site_name" content="News name" />
<meta property="og:locale" content="en_GB" />
<meta property="article:section" content="Africa" />
<meta property="og:url" content="URL of the article" />
```

Linked Data is presented as being a set of best practices when coding metadata [BHBL09]. It relies on RDF information present in files to create a data structure on the Web and to connect information from different data sources (a network). Its knowledge base contains RDF links that can be crawled, similarly to HTML links [BHIBL].

Json-ld is a lightweight JSON interface for Linked Data endorsed by Google. This interface is equally used as Open Graph by news agencies. Since Open Graph has a nature of "social", by Facebook, some information is only embedded in Json-ld. Some examples of information described in JSON-ld:

```
<script type="application/ld+json">
   {   "@context": "http://schema.org"
       ,"@type": "Article"
       ,"url": "url of the article"
       ,"publisher": { "@type": "Organization","name": "News Organization"}
       ,"headline": "Headline"
       ,"articleBody": "Text summarizing of the article"
       ,"image": {"@list": ["image1","image2"] }
       ,"datePublished": "Date of the punishment"}
   </script>
```

For a complete extraction of metadata, from news publications, information provided by the Open Graph protocol and Linked Data should be considered.

2.4.2 Metadata in a news aggregation system

For a complete and robust news aggregation system, there must be control over which articles are accepted to the archive. The interpretation of metadata from news articles is essential to provide a control system to news aggregation systems. This data allows the system to detect nonrelevant articles by its publication date, source, topic, among others, resulting in a collection of curated news publications.

2.5 Conclusions

With the research and study, presented in this chapter, relative to available methods to design and implement a news aggregator system, it is possible to draw a few conclusions regarding the steps to take when building this system.

A news aggregator system can be divided into two distinct steps: collect the data and prepare the data to be used.

To collect news articles from the news websites, a web-crawler can be used to explore, index and save the pages available of a given organization. This method has the advantage of providing with a wider exploration range: since it can access and explore all the links in a web page, its reach is beyond the website's entity.

This advantage can also be perceived as a disadvantage since a tighter control is needed to not end up with a database too sparse in terms of its content and sources. Meta-data present in news articles can serve as a control mechanism to keep the web-crawler inside certain bounds (being it the source of articles, topics or publishing dates).

Feeds of information provided by news entities, RSS feeds, deliver information posted by this organization in a structure and systematic feed.

RSS feeds can be used singularly as a method of collecting news articles, but then again for a more complete and rich database, a two-part system using feeds and web-crawlers can be implemented. This system would rely on information provided by the RSS feeds to collect the "first layer" of news articles and then use a web-crawler to explore this same articles.

Although RSS feeds and web-crawlers can aggregate news articles web pages, web scrapping techniques are needed to extract the relevant textual information from this source-files, such as title, the body of the article, author, among others. These practices allow the system to only store relevant information and ignore user interface, publicity, and clutter that might be present in the files.

Chapter 3

Information Extraction and Similarity

In this section, we present relevant information and projects related to information extraction both from news articles and social network messages, as well as measures to assert the similarity of content between texts.

3.1 Information Extraction

In recent years, there has been a proliferation of information on the Internet, from news articles to research papers and social networks. Most of this information is usually transmitted in an unstructured form, that is, through free text documents. With the growth of unstructured data on the Internet, there is and increasing interest to automatically process information and exploit the knowledge behind such data [CL96].

Information extraction (IE) is the task to identify concepts or topics, ignoring irrelevant information present in the text or data, originating structured information from free text. Identification of entities such as persons, groups, organizations, relationships, numerical, temporal and geographical references provide for richer queries and linkage of knowledge.

Because texts do not contain all the information regarding referred topics, information might be implicit. Besides, there are several ways to approach the same topic.

Information extraction should focus on less sophisticated tools and not try to emulate the cognitive comprehension of the human brain, since it is still impossible, from a technical point of view, to compute all possible relations and interpretations of the respective semantics.

Since information extraction creates a structured representation of information, it is expected to have pre-defined structures, where each object can be associated with attributes. This allows the technique of information extraction to be used in different texts or contexts, and populate a database with the information extracted.

3.1.1 Information Extraction Tasks

Information extraction is a collection of several tasks that creates a structured view of unstructured data, to transform free text into machine readable information.

While entities and relationships to identify in a given text were traditionally coded manually, this process can become tedious and cumbersome with the amount of data and topics that we want to cover [CL96]. This leads to the development of extraction systems capable of learning rules from a set of examples, and later to more robust systems to be used in noisy data. Approaches to the extraction of relationships using Hidden Markov Models (HMM) [AG04] [BMSW97] , "a finite-state machine with probabilities on the state transitions and probabilities on the per-state word emissions" ([McC05]) became widely used in the 90s. This method was then surpassed by Conditional Random Fields (CRF), used not only for "segmentation and classification but also normalization and deduplication, using models beyond just finite-state machines" ([McC05]).

Preprocessing unstructured data The first phase of information extraction is the isolation of the several elements of a free text or article. This separation of the text in several elements is part of the preprocessing pipeline of any information extraction system.

A sentence analyzer and tokenizer divides and identifies tokens (words, numbers, punctuation) from the text, using as delimiters such as spaces, commas and dots. After dividing the text into individual elements, the system will then categorize these elements.

Part of speech tagger or POS is used to identify the grammatical category of tokens from a fixed set. These categories range from nouns, verbs, adverbs, adjectives, pronouns among others.

A parser then organizes the constituent tokens in a tree like structure, depending on their grammatical information. This is a crucial part of information extraction since extraction of entities (usually nouns) relies heavily on the preprocessing technique.

Named Entity Recognition This task allows for the detection and identification of entities and type that might be present in the text. Predefined types of entities extracted are usually: organizations, persons, place names, temporal expressions, numerical expressions, among others.

The application of entity extraction is two-folded: the presence or not of entity types can be, by itself, used as a feature of the message; the correlation between the entities extracted can be used to determine the relevancy of a message and associate it with real world events.

In addition to detect and collect entities present in the text, storing this information is crucial to be able to compare several texts. The storage of structured data should be predefined, so it is central to this task to identify types of structures that can be used and their advantages.

One approach to store Named Entity Recognition (NER) entities is to simply store the number of entities present in the text of each given category (organization, person, local among others), Table 3.1. This type of representation can be used when comparing texts of similar nature and size, but the downside to this structure is losing information about the entities.

Table 3.1: Entities' representation by theoccurence frequency in the text

Local	**Person**	**Organization**	...
2	1	1	

Storing the identified entities in the correspondent categories is another approach that allows us to compare different types of texts (Table 3.2). Although there is no loss of information in this structure, comparing fields with one value against fields with two or more values can result in bad linkage between texts with similar information

Table 3.2: Representation of entities by their values in categories

Local	**Person**	**Organization**	...
USA, California	Barack Obama	NATO	

A more complete and rich structure can be applied to store the entities recognized in the text through NER techniques. This structure would allow representing more precisely the information present in the text, subcategorizing the entities in terms of their nature (Table 3.3). In this way, a place name can be a continent, country or city. A person's name can be linked to their environment such as political, musical, among others. This type of categorization can be achieved by providing to the system dictionaries, like the list of countries and cities, or current political figures.

Table 3.3: Representation of entities by their values in their categories and subcategories

Local- Country	**Local - State/City**	**Person - Political**	**Organization- Worldwide**	**...**
USA	California	Barack Obama	NATO	

Relation Extraction Is the task of detecting predefined relationships between entities in the text. This information is relevant because it allows extracting relations between isolated entities [SCM99]. Correct identification of relationships between entities is correlated to the correct segmentation and selection of entities from text. It is important to notice that a relation extraction technique is only as powerful as its entities extraction technique.

Relation extraction suffers from the fact that the accuracy can vary with the domain of the text, and might require a deeper and more complex understanding of the language and topic [McC05].

Considering that relationships must be predefined, they are usually used for focused information extraction. One example of a focused information extraction would be to extract the items and prices present on a commerce website, where the relation "price of" can be extracted between a product name and a numerical value (price).

A generic formulation of this relationships can be used for the detection of more relevant information in a broad context. Since the context of this work is comparing information between

news and social network messages, extraction of relations between entities might have to be looked at in a more generic form. Examples of relationships usually present in this type of text is location. The "location of" would be able to identify an event on a news article and social network message, with the correspondent name of the place.

Event Extraction The extraction of events in free text allows for more detail in the structure of. The task usually tries to identify entities and attributes that best relate to the questions: "who", "what", "when", "where". More attributes can be added to enrich the knowledge structure, but can also be damaging in generic contexts, as they might not apply to most contents.

3.1.2 Information extraction from news

In this dissertation, as previously stated, we will be presenting a system to replace manual labeling of social network text as newsworthy or not. This assertion will be possible by comparing information present in news articles' text with information present social networks' text. Therefore, it is important to explore the similarities and differences of these two types of texts, and what type of information extraction techniques can be used.

3.1.2.1 Journalistic text

The journalistic text is a form of communication of selected information on events, presented in the form of news articles or programs. The text is usually well structured and independent, that is, it usually presents all the information needed to understand the event or information that is trying to be transmitted.

Although there have been efforts to standardize the delivery of structured information present in news articles online, through metadata, information extraction of this contents still proves to be a difficult task to achieve. Due to the size of news articles, it is important to combine several information extraction techniques for a more complete extraction of information.

3.1.2.2 Extract information from news articles

Information extraction from news articles can rely on the content of the text or on the metadata related to the news article.

Text based information NER can be used to create a vector of features that describes a news article. The features can be related to the presence or not of certain entities, the times it appears in the text or the presence or not of types of entities.

Stanford Named Entity Recognition [Sta17] is a state-of-the-art tool for NER, as it can provide with the identification of organizations, locations and persons. An example of this tool can be seen in Figure 3.1

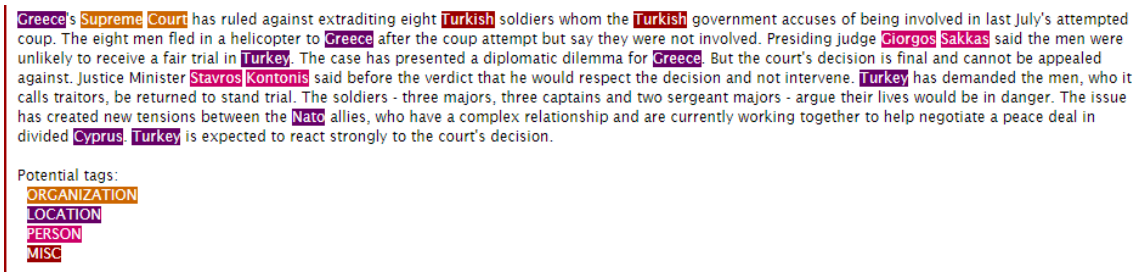

Figure 3.1: Stanford Named Entity Tagger online experiment, [Sta17]

Keyword extraction from news articles and a summary is provided by a state of the art Python library called Newspaper [New17] already referenced in Section 2.1. Keywords are the top 10 terms with the highest number of occurrences (frequency) in the article's text. The summary is the top 5 ranking sentences, depending on the frequency of each keyword in that sentence

Since the journalistic text is usually related to real world events, the extraction of this same events is an essential part of information extraction from this texts. As pointed out by Petrovic [PO13], the identification of events can be achieved by clustering news articles based on their similarity. Another approach would be to identify the entities that can respond to the questions of ***who*** *did* ***what when*** *and* ***where*** .

Metadata based information Metadata information can provide with the title, news organization, topic and summarization of the article. This information is what it might be considered relevant in a news article, as it should give enough information to understand the context or events.

One approach to use this information is to provide weights or values to the vector of extracted entities, other than just the frequency in the text. It can also be used to detect similar/duplicate news articles [MJD07] or to discard untrusted sources.

3.1.3 Information extraction from social network messages

3.1.3.1 The rise of news from social networks

With the rapid growth of social networks in the recent years, the need to filtrate and extract relevant information from this vast pool of data has also grown. Social networks have allowed for everyone who has access to these platforms to express in real time, be it to friends or to broadcast information to a huge community.

The ease of use of social networks and their real-time nature has encouraged users to use these platforms to share what is happening around them. In fact, social networks such as Twitter and Facebook have been the main stages of breaking-news both by official sources or by first-hand observations [CMP11].

Journalists and news editors have found useful the usage of social networks, not only for detecting breaking news but to extract new information. Journalists review and analyze responses

and reactions to the publication of news articles, to follow-up new information that might be relevant to the stories [ŠTP+13].

Besides news, information extraction in social networks has proven to be useful to rescue people in emergency events [Vie10], to prevent calamity incidents, such as earthquakes [SOM10], by monitoring keywords such as "earthquake" or "shaking", or even to track epidemics [LDC10].

Although extracting information from social network messages is extremely relevant, it carries additional problems, when compared with information extraction from the news. The high volume of data (real-time) and noise of messages can compromise the performance of information retrieval systems used in news articles [POL10].

One approach to information extraction from social network messages is a word focused one. For example, to look at the presence of numbers in a message and associate them with a range, and identifying each range as a feature. The types of entities present, or not, in the text can also be viewed as features of a message [AVSS12].

Another approach is to use term frequency-inverse document frequency (td-idf) to measure how relevant a word is in a message, that is, its relevancy increases with the number of times the word appears in the text [POL10]. Although this approaches might represent a start, in the extraction of information, there are complementary approaches to take full advantage of the information present in social network messages.

3.1.3.2 Extracting events from social networks

The purpose of information extraction from social network messages is not only to extract atomic entities from text but also to build up richer and more complete structures such as events. Event identification and extraction leads to a deeper understanding to the automatic system and allows us to gather and filter information relating to specific topics. This becomes pertinent when detecting breaking news or when gathering new information related to emergency events, as an example.

Event extraction in social network messages has been a recurrent research topic in recent years and can be grouped by the nature of events that are being extracted [SOM10, BNG11a] or by the techniques used [AVSS12, ZCH+14, RMEC12]. Event extraction can be as generic as breaking-news or more focus to cultural events, calamities detection, accidents, among others. Approaches to event detection are usually achieved by extracting features from individual messages or clustering the messages (as explained in the following paragraphs).

Since social media messages are usually very short, its content might not contain all the aspects of an event (location or entities involved). This becomes a problem when clustering messages based on their word-similarity, possibly resulting in clusters referring to the same aspect of an event. It is important to consider the semantic information conveyed in a message and not only its words independently.

Temporal features Since social network messages are usually posted in real time, temporal features are central to event identification.

One approach proposed by Becker [BNG11a] is to aggregate the number of messages relating to a frequent topic or term in hourly bins. This will capture any deviation to the flow of expected messages and become a feature to the messages that will help the system better capture events. One example can be seen in Figure 3.2, where it is presented the number of messages collected with the term "valentine" by the hour in a time frame of 72 hours. It is clearly visible the growth in the number of messages as the date gets closer to Valentine's Day.

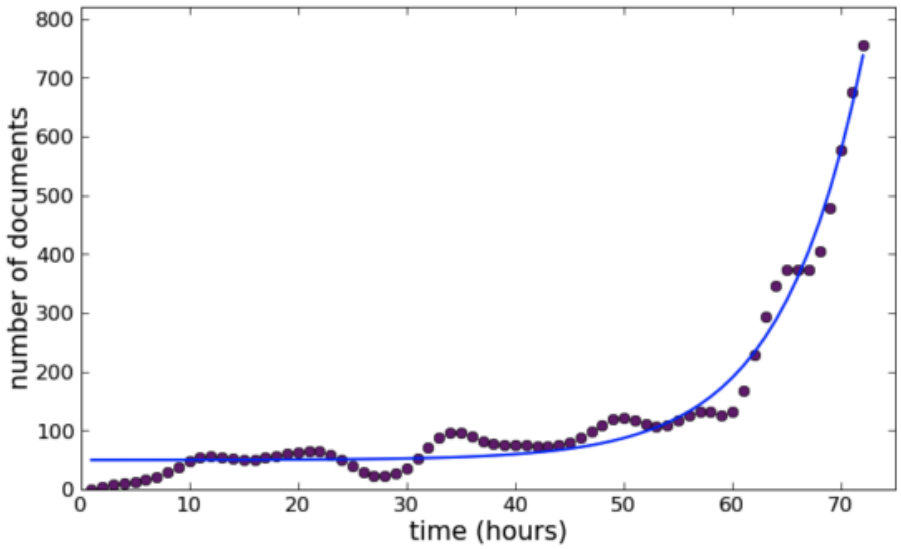

Figure 3.2: Documents per hour with the term "valentine" from 72 hours prior to 2 p.m. on Valentine's Day, Twitter, from the article [BNG11a]

Entity feature NER in social network messages is key for information extraction. One tool that can be used in the field of social networks for this extraction is Stanford NER, as presented by Agarwal [AVSS12]. Entity recognition is pertinent since a message related to an event or news is more likely to refer to some of its entities.

Clustering of messages One approach to the clustering of social network messages, based on events, was proposed by Petrovic [POL10]. In this system, messages are represented as vectors, where each feature or coordinate represents the frequency of a word or term in the text. Each new message is then compared with the processed messages and its similarity is measured.

If the similarity between the closest message is below a certain threshold, it is created a news cluster or event. Otherwise, the message is associated with the group of the closest messages. This

method was also used by Petrovic [PO13] to create clusters both for events from Twitter messages and newswire articles.

3.2 Similarity Feature vector

3.2.1 Similarity

The similarity is the measure used to compare the likeliness and resembles between objects. While this concept might be relatively easy for humans to understand, machines must compare numerous values extracted from objects to be able to assess this measure.

Before calculating the similarity between object, the system must first extract the features that characterized the object, usually in the form of a vector of features. For a system to be able to assess the similarity between objects, the same set of features must be extracted from both objects. This can bring some difficulties when trying to assert the similarity between two different types of objects.

Since one of the tasks in this dissertation is to assess the similarity between news articles and social network messages, it is important to define a set of features that can be extracted from both sources.

3.2.2 Similarity between vectors

Cosine similarity is the measure of the cosine of the angle between two vectors. Its value can vary from 1 to -1, where relatively similar objects score values closer to 1, unrelated when closer to 0 or negative relation of the features when closer to -1.

One of the applications of cosine similarity is when comparing sentences or documents, considering the several entities or keywords referenced. The representation of documents or strings as vectors is called a vector space model. A representation of a vector space model can be seen in Figure 3.3, where each axis corresponds to an entity or term, and the vector a sentence.

Since cosine similarity only considers the angle of the vector and not its length, a document where, as an example, the word “blue” appears 10 times and another document where it appears 400 times, the angle between the feature vectors would still be small. In this way, cosine similarity does not look at the frequency of a word in a vector space model, but to the topics and entities referred. This is extremely helpful when comparing texts of different lengths such as social network messages and news articles.

An approach to the similarity between social network messages and news articles was presented by Petrovic [PO13]. The purpose of this research was to analyze if Twitter messages could break news before the news-wire. For this, Twitter messages and news articles would have to be matched, to compare the publishing time of both. Twitter messages and newswire articles were

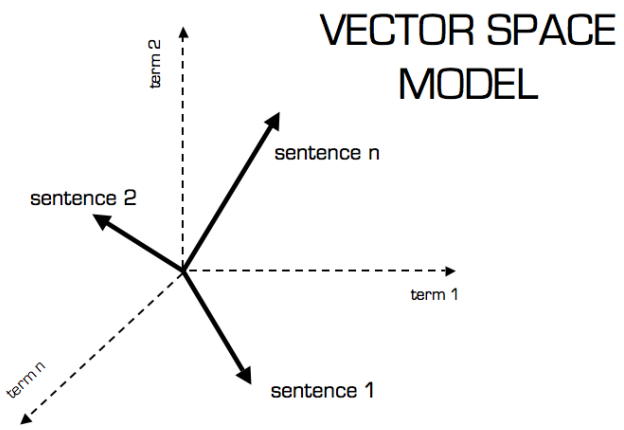

Figure 3.3: Representation of sentences by a feature vector in a vector space model.

clustered by events [POL10] separately. Each cluster corresponds to an event or news reported in each stream of information. After the clustering of events, the focus was on aligning events reported on both Twitter and newswire.

To detect the events covered by both sources, clusters from one source were aligned with the cluster from the other source with the closest cosine similarity (Figure 3.4). This then allowed for the researchers to compare publishing times between the aligned pairs and detect tweets published before the equivalent news pieces.

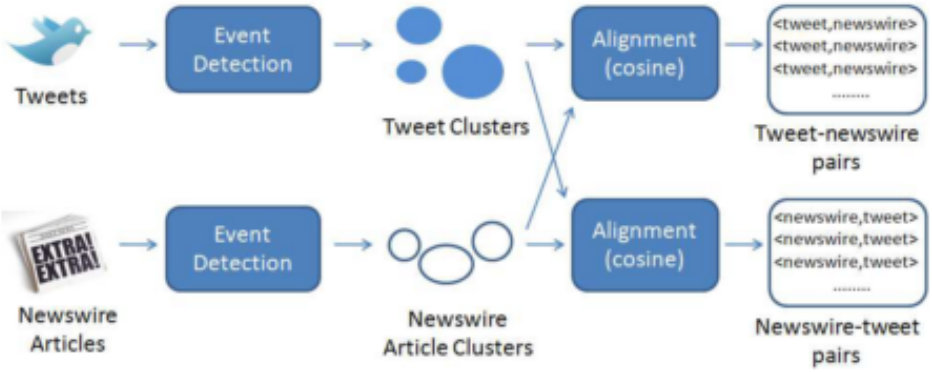

Figure 3.4: Methodology for automatic generation of tweet and news wire clusters pairs, from [PO13]

One of the conclusions drawn from this experiment was that "1% stream of Twitter contains around 95% of all the events reported in the newswire" [PO13]. This is an enticing result that supported the development of this dissertation: the majority of news content is somehow present in social network messages.

Another use of cosine similarity was presented by Sankaranarayanan et al [STS+09] and

Becker et al [BNG11b] as a measure of distance between social network messages to create clusters.

3.3 Conclusions

Several conclusions can be drawn from the study presented on information extraction and similarity measures.

Regarding the extraction of information from text, this can be divided in two main phases: preprocessing and extraction of information.

- It is important to design and implement a rich and comprehensive pipeline of preprocessing techniques to be able to extract the maximum information possible from the text.
- Extraction of information should be predefined, to be able to assess, compare and link information between two different types of texts: news articles and social messages. These aspects regard the type of information that should be extracted, as well as the storing techniques.

The dissimilarity between journalistic text and social network text can bring problems when extracting information from this two different types of texts.

Seeing that journalistic text is more complete in its content, extracting information from this type of text will prove to be richer than social network messages. Due to their real-time nature, the short snippets of information coming from social networks usually lack information that would be present otherwise. Comparing both texts will require a delicate curation of feature extraction techniques to use.

Extraction of information will be prioritized over quantity, that is extracting more information from the entities and relations present in the text over the number of times the entity is present. This is a crucial conclusion when studying information extraction and linkage between journalistic text and social networks messages: since the size of both texts is widely different, the information provided by the content is more relevant than its cardinality.

To compare and assert a degree of likeness between information present in news articles and social network messages, two types of approaches can be used: similarity measure between feature vectors or clustering of text.

- To compare the information present in two texts such as a news article and a message, and consequently their vectors, a measure must be used to assert how similar both vectors are.

  Cosine similarity measures the angle between two vectors and can provide with a degree of similarity between 1 (identical vectors) and -1 (reversed vectors). Using this measure,

a threshold of similarity must be chosen to assert whether a message's information vector should be categorized as news-worthy or not, depending on its similarity with news article information vector.

- Similarity measures can also be used to cluster vectors of features, based on how similar those are in the space model. The clusters are categorized by similar vectors, and therefore similar information. It is then possible to determine a degree of similarity in which a vector should be added to a cluster.

  To determine if a social network message should be categorized as news-worthy, the clustering method can be used: if the message is added to a cluster of news articles, then its information is similar to those same articles and therefore news related.

Chapter 4

Automatic assessment of accuracy

In this chapter, the work developed in the dissertation will be presented. In Section 4.2 and Section4.3, the problem and proposal for this dissertation are presented. In Section 4.4, the overview of the developed work is laid out. In Section 4.5, the aggregation system of news and messages is described. The information extraction pipeline is presented in Section 4.6. In Section 4.7 several experiments are conducted in order to assess the precision of the matching system and elaborate the best matching method. In Section 4.8, the performance of the system on calculating the accuracy of a classification system is evaluated. The conclusions of the work are presented in Section 4.9.

4.1 Introduction

From casual conversations to important and lifesaving publications [SOM10, PO13], social networks have become one of the main feeds of information and content in the world [].

The impact that these new means of communication have in the propagation of news [PO13, Mur11] have attracted researchers to develop methods to filter information from the vast pool of messages shared in this media [STS+09, CMP13].

4.2 Problem definition

One of the main obstacles when studying and conducting research on social networks is the labeling off the studied data. Labeling data is essential for the development of many research projects developing classification models, as it is with this information that any model can compute its accuracy and provide insight on the performance of the work being developed.

The labeling of data can be achieved by two distinct processes: either my manual effort, where humans classify the data, or by and automatic system, where computers are assigned the task of classifying data.

Research in the field of social networks usually relies on manual labeling of data [PO13, CMP13], to be able to assess the accuracy and precision of the work being developed.

Whilst manual labeling has the advantage of low implementation cost, supporting this type of categorization becomes unbearable to any project, being it monetarily, time spent on the labeling or the accuracy of the work. These problems are amplified when the classification evolves into subjective terms, as is the case of news relevancy.

Automatic labeling, while preferred, requires a more complex architecture and design of the system to be able to perform this task as adequately as manual effort. For this same reason, it is an important step to make in this field in order to to enable a sustainable venue for research projects working on social networks [FSF16, PGFA17].

The purpose of this dissertation is therefore to implement an automatic system capable of labeling social network messages and assess the accuracy of a classification model with no added costs, up-to-date data and with enough precision to be a considered a valuable tool for the projects. This can be considered as the main contribution of this dissertation to the scientific community: provide a performance evaluation automatically.

This dissertation was developed in the context of the research project "Relevance Mining Detection System" (REMINDS/ UTAP-ICDT/EEI-CTP/0022/2014), a project where the goal is to develop a classification model capable of classifying social network messages as relevant or not. In Section 4.2.1, the disadvantages of manual labeling will be explored, especially applied to the REMINDS project, and the applications that this work can bring to research. In Section 4.2.2, the term news relevancy will be discussed and how to define it.

4.2.1 Project REMINDS

The development of the research project REMINDS has been a continuous attempt at developing a classification model capable of categorizing social network messages as being newsworthy or not. The project work-flow, as seen in Figure 4.1, can be divided in three steps: collection of data, labeling of data and development of the classification model.

To obtain a dataset to train and test the classification model, social network posts, comments and tweets are collected and filtered from Facebook and Twitter. This extraction is bounded to a time-frame and topics, to restrain the pool of messages and have consistency in events and subjects referred.

Using a micro-tasking platform, users are paid to perform the labeling of the social networks messages extracted, classifying them as newsworthy or not.

Comparing the labels attributed by the model with the ones from the manual labeling, the accuracy of the classification model can be computed. The accuracy guides the development of the classification model, in the way that serves as reference on whether the model is being successful in its classification task or not.

The human component in the labeling of social network messages brings some problems to the research. People can understand or have different views on what is newsworthy, since some

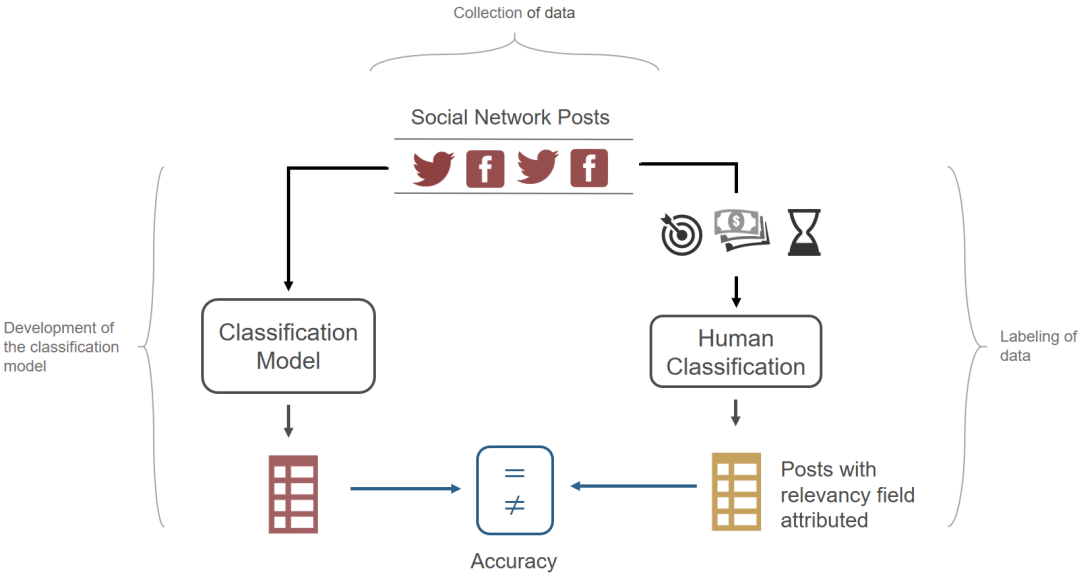

Figure 4.1: Workflow of REMINDS with manual classification of Social Network Posts

take into account their beliefs and political alignments when labeling posts. Users of the micro-tasking platform usually try to fill in the labeling as quickly as possible, possibly labeling the texts randomly or not with great accuracy [FSF16].

Beside incorrect labeling, the fact that this process is remunerated, scaling the classification can become impossible to support due to its costs. The fact that this process is not automatic, the real-time nature of social network messages losses its strength and becomes outdated. Monetary costs lead to less periodical labeling tasks, which aggravates the outdated state of the labeled data.

Therefore, it is necessary to design and implement an automated system capable of achieving the same goal of assessing the accuracy of the model, with no added costs to the research project and human intervention in the labeling phase.

4.2.2 News relevancy

One of the main problems regarding human classification of social network messages on their news relevancy is related to the subjectivity of this term. It is important to define this term, in order to implement an automated system capable of reproducing this definition, labeling posts regarding their news relevancy.

From here on forward, let us refer to social network messages as messages and news articles as news.

News relevancy or newsworthiness can be described as the property of information that can be used to generate journalistic content. This is not only bounded to the content but also the timing of the information, meaning that content that was already available long before it was shared is not considered to be relevant.

The definite assumption, **Assumption 1**, that will be taken is that a message is considered news relevant if its content or information is present in a news article. Therefore, if *Assumption* 1 holds, we say the message is 'relevant'.

$$Assumption1 : \sigma(T_{Message}) \dot{\cap} \sigma(T'_{News}) \neq \emptyset \tag{4.1}$$

The focus of our research is now on how to define the transformation of the text (σ) and what to consider to be an intersection between messages and news ($\dot{\cap}$).

Whenever *Assumption* 1 holds (intersection between the message's transformation and news's transformation is not empty) it is assumed a certain degree of relevancy, meaning that the message is related to one or more news articles.

Transformation σ corresponds to the extraction of information from a text, allowing the system to store and access this information in a standardized manner for texts of different natures and sources (messages and news).

Intersection $\dot{\cap}$ allows the system to compare both transformations of the text (information) and ultimately decide on the newsworthiness of a message.

This definition of news relevancy will allow for an implementation of an automated process to label messages using the information present in news articles.

Since the system will be using news articles as the basis for the news-relevancy classification, these will dictate the correct or incorrect labeling of social network messages.

With the growing usage of the internet as the main source of information for most people, untrusted outlets have been gaining ground, sharing false and rumorous information to distrust and deceive the population. To develop a labeling system in which we can trust, we need to make sure that the news content being collected is true and coming from trusted sources. If the system collected all news pieces available, this would probably contain false information, or fake-news, mislabeling the messages. Therefore, it is essential to carefully curate the sources of these articles to gather reliable information.

4.3 Proposed solution

We will be describing the proposed solution to the problem of manual labeling of social network messages. This solution will be an automatic system capable of labeling messages on their newsworthiness, assessing the accuracy of the classification model developed in REMINDS. The solution provides an alternative to manual classification of messages to REMINDS and other research projects.

As a substitute to the manual labeling of social network messages, the system will be using the definition of news-relevancy as seen in *Assumption* 1. The high similarity of events being discussed and presented in messages and news will allow the system to label the data on heir

newsworthiness. An architectural view to the proposed solution in the case of the REMINDS project can be seen in Figure 4.2.

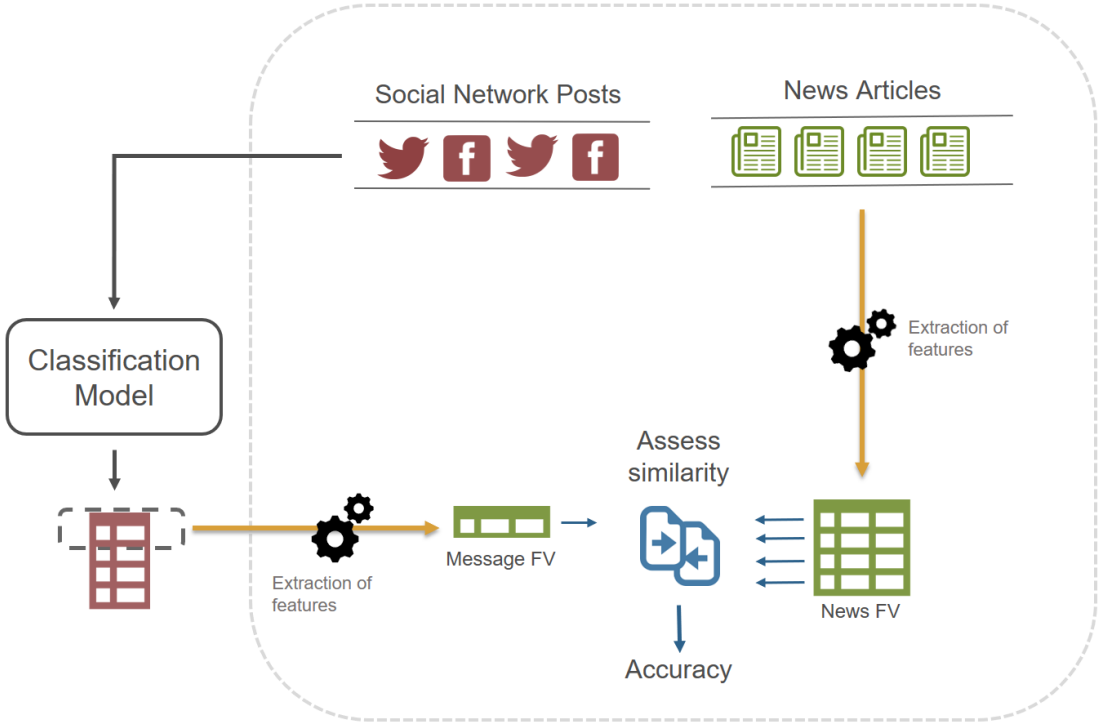

Figure 4.2: Workflow of REMINDS with proposed system for automatically calculate the accuracy of the system

4.3.1 Architectural solution

The proposed system provides a database with information extracted from newswire articles and messages, as well as a method for evaluating the similarity between the database entries of both.

The database of news was built using a news aggregator that collects and extracts news articles and publications from several newswire websites. The collection of news articles and publications is restricted to similar time-spans and topics used to collect the messages from social networks.

There is a restriction to the number of articles explored and stored, as well as newswire sources used. Without these limitations, the database could become unmanageable, due to its size, and contain information that could be considered not reliable.

News and messages are stored in the database in a systematic and strict manner, represented by a feature vector. Relevant features of the text and metadata are extracted and stored in tuples such as locations, entities involved, time of the events and report (transformation σ of the text referenced in *Assumption* 1, Section 4.2.2).

The selection of relevant information and features to be used was an obstacle overcome in the development of the system. Social network messages experience a similar feature extraction

model to store equivalent features, so that similarity between messages and news articles can be calculated.

To assess if the classification model has correctly classified a social network message as relevant, the automatic system determines if the information of the message is present in any form in the database of news. A method for comparing the message's feature vector and the news' feature vector was developed (intersection $\cap$ between the information from the message with the information from the news, *Assumption* 1 in Section 4.2.2).

The proposed system allows the accuracy of the classification model to be assessed automatically with no added costs.

Another advantage of this design is the ability to continuously introduce new data from news articles, maintaining the database updated.

4.4 Automatic labeling and accuracy system

The proposed system presented in Section 4.3 achieves to answer all the problems that resided with the manual labeling of messages in the research project REMINDS. However, the envisioned system could have other applications. With this in mind, the built system was developed independently from the project's code to enable its use in other research applications or programs.

The independent built system to automatically label social network messages can be separated into four distinct phases, Figure 4.3:

1. **Data collection:** since the purpose of the system is to label messages as being news related or not, these need to be collected in the first place. News will also be collected to help in the classification of the messages, serving as the knowledge base to this tool.

2. **Information extraction:** after collecting the data, the system needs to extract information to be able to link and associate information present in a social network message's text to a news article's text: metadata and information present in the text that will allow pinpointing a message to a specific event in the news.

   Specific features are extracted from both sources of text, to enable a fair and comprehensive comparison between the information present in each text. The set of features chosen will constitute of a vector of features used to represent the relevant information present in the content.

3. **Matching of news with messages:** in this stage, the system already extracted information from both news and messages and will match the message with the most similar news.

   Comparing information present in social network messages and news articles, and inherently their feature vectors, is to measure how similar those vectors are. Techniques on how to compare and assess the similarity between both texts were tested and applied to the development of the system.

4. **Message classification:** the messages to which news have been associated with will be classified as news related. If no news article has been matched with a message, this will be classified as not news related. This step of the system will essentially replace the manual labeling of news.

   With the classification of the messages, the system can now compare the results obtained by the labeling tool with the predictions of the classification model being developed, assessing its performance automatically and without human intervention.

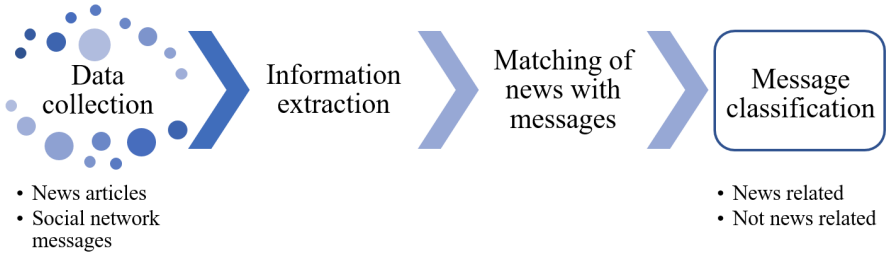

Figure 4.3: Representation of the system

4.4.1 System evaluation

Besides implementation, an imperative step in developing the labeling and precision system is the validation of said system. A test phase is expected along the development and implementation of the architecture to validate the system. Ideally, every section would have a specific evaluation technique.

While it might seem senseless to evaluate a system built to assess a precision of another system, this is a necessary step to obtain the accuracy of the work being presented.

For data collection and information extraction, the validation of these phases was made manually, only examining if the system performed correctly the intended functions.

In order to validate the matching mechanism of news with messages, two experiments were conducted in Section 4.7.2 and Section 4.7.3.

To test and evaluate the message classification method and the performance computation of the classification system, another experiment was performed and described in Section 4.8.2, validating this section.

4.5 Data collection

The collection of data is the first phase of the developed system. At this stage, the system will collect news articles to serve as the knowledge base in which the classification of the collected messages will be based on.

The aggregation system for news articles is described in Section 4.5.1 and the collection of social network messages is presented in Section 4.5.2.

4.5.1 News collection

The system needs to be able to collect a vast number of articles from different organizations. This is the data on which the classification of messages is based.

The collection of data from news websites can be achieved either by scrapping the information present in the code (web scrapping) or using Rich Site Summary (RSS) feeds, provided by the organizations, as referenced in Section 2.5.

Although web scrapping of web pages would be possible, it would not be sustainable to grow the number of sources, as it would require the web scrapping system to adapt to all the different websites [RGSL04]. RSS feeds, on the other hand, are widely used and standardized. Therefore, RSS feeds were chosen as the aggregation method of news articles.

A concern when developing the news aggregator was the reliability of the sources and collected data. A selected group of news organizations, mainly based in the United States were chosen to contribute to the collection of news. These handpicked organizations are well-known sources of credible information, meaning that the collected data should be reliable. The list of organizations can be found in the Appendix A, Table A.1.

Since these organizations provide several RSS feeds, depending on the topic of the article (business, entertainment, health, politics, science, sports, technology and world news), they allow the database of news to be diverse and cover a broad range subjects of interest to the public. The final list of RSS feeds was comprised of 134 feeds from 13 different news organizations, as presented in Table 4.1.

Table 4.1: Number of RSS sources by topic

Topic	**Number of RSS services**
World	28
Business	23
Entertainment	23
Sports	19
Technology	15
Health	12
Politics	11
Science	3

Once the system is not designed to be focused on a single topic, the unbalanced number of RSS services should not be a handicap to the system. The aggregation of news only relies upon provided RSS links, so expanding the sources is easily achievable by adding the new URLs to the list of RSS links.

4.5.1.1 News collection system

To process each of the RSS links and store the information provided by their feed (link to the article, the headline, the description of the news, and the date of publishing), a script in R [R C12]

was developed.

Although the information provided by the RSS feed was relevant to the article, some data would still be missing, such as the text of the article itself. To address this issue, the newspaper library [New17] in Python was used to extract all the data possible from the article, using its URL: the text of the article, a summary, and keywords.

Keywords are the top 10 terms with the highest number of occurrences (frequency) in the article's text. The summary is the top 5 ranking sentences, depending on the frequency of each keyword in that sentence [New17].

The integration of the newspaper library allowed for a more comprehensive and complete extraction of data from every article.

The system of news aggregation continuously collects all the information present in each RSS feed and stores the data in a SQL database. These are the features collected and stored each article: **URL (RSS), link (article), topic, headline, description, date, text, summary and keywords**.

Since there are no real constraints to the frequency to which the system pools information from the RSS links and aggregates the data, the assemble of news is performed in a continuous loop, processing one link of RSS feed after another.

4.5.2 Social network messages collection

The collection of social network posts is as important as the collection of news articles for the development and testing of the envisioned system. The messages will be the labeled data used to assess the accuracy of the classification model.

Facebook and Twitter were the social networks selected to extract posts from, due to their wide user base and easy accessibility to the data, provided by their APIs [Twi17] [Fac17] .

It is important to define the sources that will contribute to the dataset of social network messages: journalistic and non-journalistic. With journalistic sources, the purpose is to collect posts that will most likely contain information present in news articles. With non-journalistic sources, the probability of a message containing news relevant content is slimmer.

The aggregation of posts can be achieved either by collecting the posts published by a given account or posts that reference a given topic. For journalistic sources, a list of news organization accounts present on Facebook and Twitter was created, so that the system could regularly collect the posts published by those accounts.

For non-journalistic sources, it would not be feasible to create a list of random users and expect some of its content to be news related. For this reason, another approach was taken to solve this problem: trending topics.

Twitter provides through its API [Twi17] access to trending topics: a set of terms and hashtags currently being used in its network by a large number of users, in a certain country or city. Using the trending topics (restrained to the US), the system can search for those terms and collect the

messages being shared. With this method, there will always be a percentage of news related posts, published by non-journalistic sources, being collected by the system.

4.5.2.1 Social network messages datasets

The collection of social network messages will be divided into three specific sets, to analyze the system's performance in different scenarios, Table 4.2:

1. Posts published on Facebook and Twitter by the 13 news organizations present in the RSS list: social network RSS dataset (**SNRSS**). This dataset will be composed mainly of posts referencing the news articles collected, so the number of news related posts is expected to be the highest of the three.

2. Posts published on Facebook and Twitter by other news organizations, not present in the RSS list (27 on Facebook and 25 on Twitter): social network news dataset (**SNN**). This will be primarily comprised of posts related to news events, but not to a specific collected article, meaning that the number of news related posts would still be significant, but lower than SNRSS.

3. A collection of posts sampled from trending topics on Twitter: social network trending dataset (**SNT**). This dataset is expected to have the lowest percentage of news related post, helping to stress the system on its accuracy with more casual text.

Table 4.2: Number of sources (accounts) from Facebook and Twitter, for each dataset, used to collect messages

	SNRSS	**SNN**	**SNT**
Facebook	13	27	-
Twitter	13	25	Trending topics

More detailed information related to the sources used to collect social network messages can be found in Appendix, Table A.1.

4.5.2.2 Social network messages collection system

To aggregate data from social networks, the system used APIs provided by Facebook and Twitter to search, gather and store posts and their information. To extract and collect messages from Twitter, R was used with the package twitteR [JG15], using the Twitter REST API [Twi17] . To extract posts from Facebook, the language chosen was Java with the Facebook REST API [Fac17].

The assemble of social network messages requires the implementation of constraints to respect quotas imposed by the social networks' APIs:

- In both SNRSS and SNN collections, the pooling of new messages is executed every 60 seconds, one account at a time, for Facebook and Twitter.

- In the case of SNT collection, it is comprised of two phases: a request of all the trending topics currently being discussed on Twitter in the United States, followed by a search of each term, every 30 seconds, with the collection of 300 tweets [JG15].

The collected data was then stored in SQL tables:

Data collected and stored from Twitter: {tweet id, user id, date, latitude, longitude, favorite count, re-tweet count, in reply to, language, text}

Data collected and stored from Facebook: {post id, user id, date, caption, message, shares count, like count, comment count}

4.6 Information extraction

To compare social network messages with news, the system first needs to extract insightful information from the text of these two types of objects. The data, both from news articles and social networks, will be run through the same information extraction pipeline, to obtain a standardized representation (tuples) of their information.

4.6.1 Entity extraction

The approach to extract information is by using entity recognition. The rationale behind this method is that texts with common entities will most likely be related to the same event.

The entity recognition pipeline was developed in Python with the Stanford CoreNLP [MSB+14]. This toolkit allows the identification of four different types of entities: persons, locations, organizations and dates.

For each social network message, the entities present in the text are extracted and stored in a SQL table with the following tuples: message_id, persons, locations, organizations and dates. While the message_id refers to the ID of the post in the SQL table, the other tuples were composed of vectors, where each vector contained entities related to the group. An example:

Text: The Federal Reserve's plans to raise U.S. interest rates gradually are aimed at sustaining full employment and near-2-percent inflation without letting the economy overheat, Fed Chair Janet Yellen said on Monday.
Persons: {Janet, Yellen} , **Locations:** {U.S.} , **Organizations:** {Federal, Reserve, Fed} , **Dates:** {Monday}

In a similar fashion, the same pipeline was applied to news articles. The SQL table resulting from this extraction was composed by the news_id, referencing the ID of the article in the SQL table, and the persons, locations, organizations and dates vectors of entities present in the text of the article.

Since social network messages and news articles have widely different texts length, we opted to collect only the entities referenced in the text and not how many times it was referenced. This

means that if a person is referenced 5 times in an article, the entry "Person" in the SQL database will only show one instance of that name.

4.7 Matching of news with messages

The third stage of the labeling system is to compare social network messages with news articles collected. As previously stated, the newsworthiness of a message is bounded to its time-frame. To ensure the relevancy of the messages being labeled, the system will only compare news published on the same day as the message. This will guarantee that the system does not match a message with a news posted long before.

In this phase, the system will compare the extracted information of one message to the information extracted from each news article, selecting the best match between these two types of objects. The selection of the best pair message – news will be achieved by an heuristic. The heuristic selection will be the study focus of this section.

If a message is matched with an article, that means its information is somehow related to the news, so its classification should be 'news related'. If the system is not able to find a match to a message, then its content is probably not present in any article, meaning that it should be classified as 'not news related'.

4.7.1 Matching method

The system compares the information of messages and news by how analogous their tuples are. This comparison generates several candidates matches for one message. The candidate's matches are then ranked by their similarity and the highest-ranking one chosen: best match (Figure 4.4).

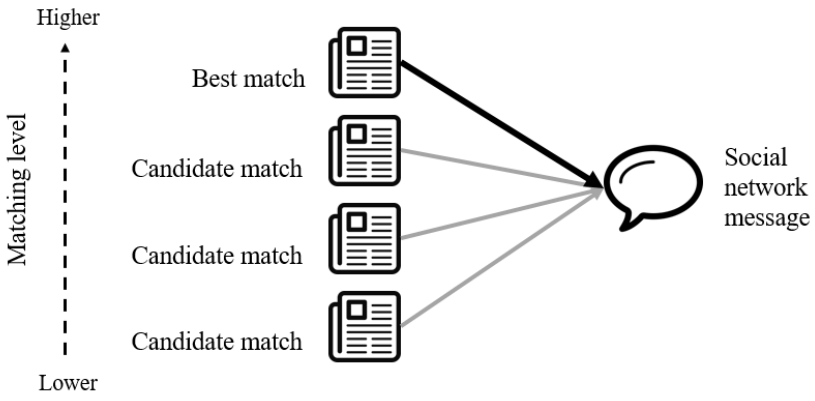

Figure 4.4: Representation of the matching system

To evaluate the matching method, some experiments were conducted. In Section 4.7.2, the precision of the system to correctly intersect extracted information between messages and news is studied, more precisely, the selection of candidate matches for each message. For this experiment,

only messages coming from news organizations (SNRSS and SNN) were used: the probably of the messages being news related is higher.

In Section 4.7.3, the emphasis is in the selection of the best candidate match and improving the precision of the system. Messages from both news organizations (SNRSS, SNN) and unofficial sources (SNT) were used to reflect how the system handles these two types of messages.

4.7.2 Candidate matching experiment

The aggregating system collected news articles and social network messages from the 8th of April through the 10th of April of 2017 (duration of 3 days), Table 4.3.

Table 4.3: Data collected

Day	**8th**	**9th**	**10th**
News	880	990	1642
SNRSS_FB	455	491	596
SNRSS_TW	932	954	1399
SNN_FB	442	388	519
SNN_TW	900	847	1346

A small sampling from each day was randomly extracted: 10 Facebook posts and 10 tweets from the SNRSS dataset and 10 Facebook posts and 10 tweets from the SNN dataset, for each day. This resulted in 40 messages per day, in a total of 120 messages.

After the sampling, the information extraction was applied to the messages and news articles. With the information stored in tables, the system could now compare and match messages with news, depending on their entities.

An important aspect when considering how to compare a message with a news is the need to get more information only than the simple similarity between messages and news. The approach chosen in this work was the idea of hits. A hit occurs when the same element is found in both objects (messages and news). With hits, it is possible to analyze the impact that each of the features has on the matching method and ultimately in labeling messages on their newsworthiness.

For each match, the table of candidate matches will be composed of the ID of the message, the ID of the matched news and four value hits, one for each group of entities (persons, locations, organizations, and dates). Besides, another value will be added: the sum of all hits of entities. The rationale behind these features is that they will help to understand what type of features the system should prioritize and how to improve the performance.

Table of matches:

news_id , message_id, h_persons, h_locations, h_organizations, h_dates, h_entities

If a message has at least one hit with a news article, then this pair is considered a candidate match and added to the SQL table of candidate matches.

In this experiment, the method resulted in 1808 candidate matches, corresponding to 69 messages (51 of the 120 messages did not have any match, being classified as not related to news). The candidate matches were then manually classified as being correct, if the message and news were in fact related, or incorrect otherwise.

From the 1808 entries, 273 candidate matches were classified as correct and 1535 as incorrect. Analyzing the evaluated candidate matches, there were 26 messages whose candidate matches were all classified as incorrect and 43 messages that have at least one candidate match labeled as correct (Table 4.5).

Since 26 messages will never be correctly matched, any analysis of the dataset should take that into account. Let us refer to these 26 messages as messages with impossible correct matches.

4.7.2.1 Matching with entities

In the following graphs (Figure 4.5) it is visible the ratio distribution of correct candidate matches, accordingly to the number of hits of the four groups of entities: persons, locations, organizations and dates. While there seems to be a tendency for a higher ratio of correctness, the number of entity hits does not give the system a clear indicator on how to classify the candidate matches.

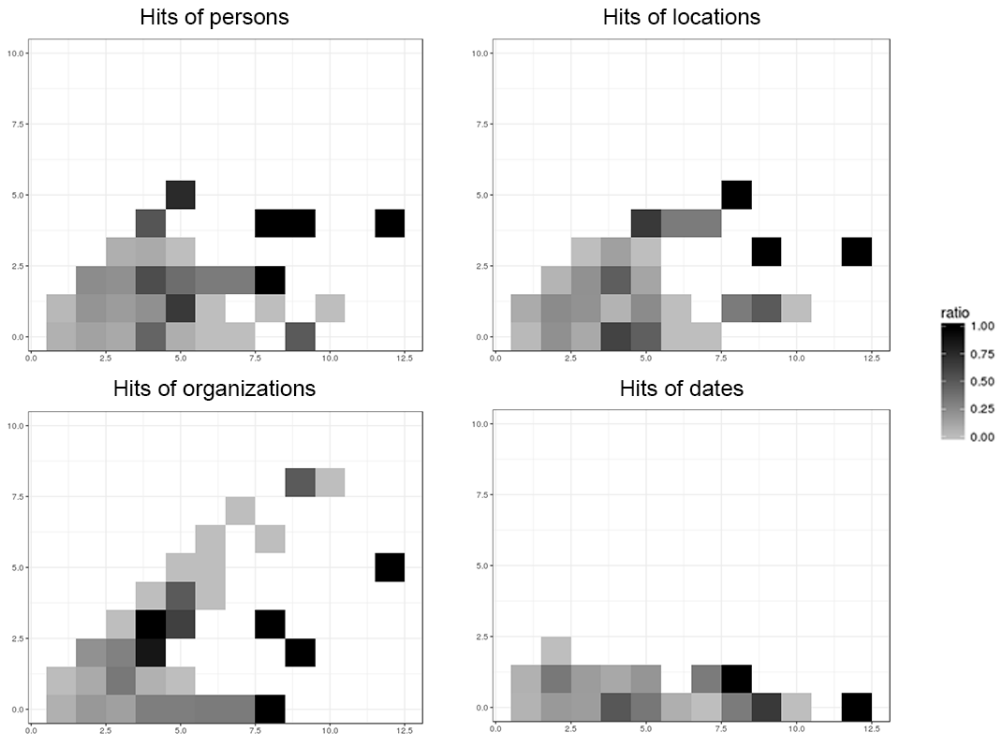

Figure 4.5: Heatmaps for candidate matches

Since there were 1808 candidate matches for 62 messages, the system should now select the best match. The first approach to choosing the heuristic of best candidate match was:

$$Heuristic\ 1 : sum\ of\ entity\ hits \tag{4.2}$$

Applying *Heuristic* 1 to the candidate matches, the system matched correctly 28 messages and 41 incorrectly (from which 26 were messages with impossible correct matches), Figure 4.6. Although this heuristic can, in fact, match messages and news with some degree of certainty, it is not a satisfactory result.

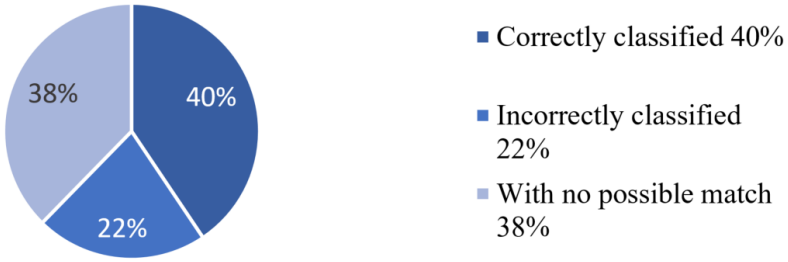

Figure 4.6: Distribution of best matches using Heuristic 1

One of the main problems of this approach is the noise that entities can induce in the system. Entities such as Donald Trump and U.S. (present 746 and 632 times, correspondingly, in the sample of news and messages) can dramatically increase the number of hits, however, that may not contribute to narrowing to a specific news event. Two examples:

SN message: Donald J. Trump has spent 10 consecutive weekends on the golf course, after criticizing Barack Obama for hitting the links.

Headline: Trump to sell attack planes to Nigeria for Boko Haram fight

Entities: Persons:5, Locations:0, Organizations:0, Dates:0

These two objects do not reference the same event or news, nonetheless, the number of hits of entities is relatively high (fourth highest score of all the candidate matches in the experiment).

SN message: Car on fire rolls up to #KFC drive-thru window

Headline: Car on fire rolls up to KFC drive-thru window

Entities: Persons:0, Locations:0, Organizations:1, Dates:0

These two objects share the same text, yet there was only one entity hit. It is noticeable the drawbacks of only using entity recognition to match social network messages with news articles.

Besides the inconsistency that entities bring to the matching mechanism, another problem arises from the conducted experiment: the high number of messages that had no viable match (26 out of 69 messages).

In the following approaches, the goal is to find new features that could improve the matching method and in the detection of posts with impossible correct matches.

4.7.2.2 Matching with keywords

To overcome the poor results of the matching process when only using entities, the system will take advantage of one of the features extracted during the assemble of news: keywords.

An example of how the usage of keywords can improve the matching of news with posts can be seen here:

Headline: Trump names Alexander Acosta as new pick for labor secretary

Keywords: committee, president ,**labor**, **names**, served, alexander, florida, **secretary**, law, acosta, civil, **attorney**, rights, **pick**, trump

Keywords allow the system to have a more detailed extraction of information from news articles. Therefore, it can be used to distinguish a specific event and correctly match news with messages with higher accuracy.

While in the previous approach, the system only compared the vectors of entities, resulting in 4 values of hits (persons, organizations, locations, dates). This improvement will add a fifth value, the hits of keywords.

The hits of keywords are calculated by the number of keywords from the article that are present in the message's text.

news_id , message_id, h_persons, h_locations, h_organizations, h_dates, h_keywords

The imposed obstacles are now the following: how to select the best match (entities vs keywords) and which threshold to choose to discard wrong matches?

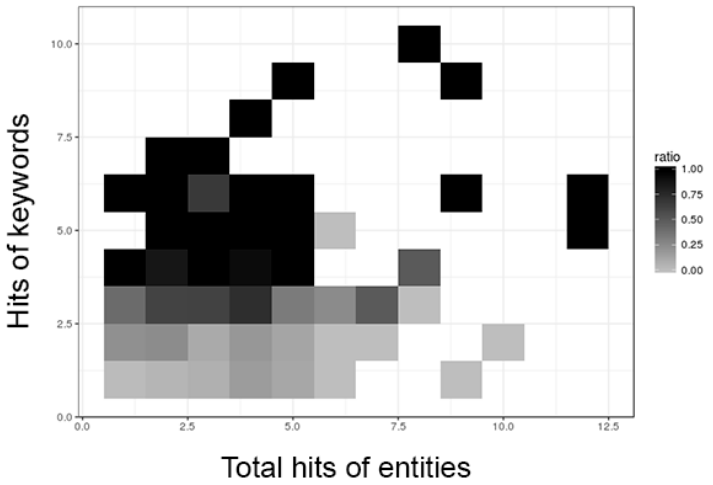

Figure 4.7: Heatmap of the ratio of correct candidate matches by the number of hits of keywords

Using the previously labeled dataset, it is possible to evaluate the differences between the hit of entities and keywords on the correctness of a candidate match, and how the system can take advantage of this new feature to improve its performance.

The hit of keywords, as seen in Figure 4.7, has a higher reliability than the hits of entities, meaning that there is a visible distinction between the correct and incorrect candidate matches.

Using the hit of keywords as the main heuristic to select the best match:

$$Heuristic\ 2 : hit\ of\ keywords. \tag{4.3}$$

Using *Heuristic* 2 the result was the following: 36 true and 33 incorrect (of which 26 are messages with no possible correct matches), Figure 4.8.

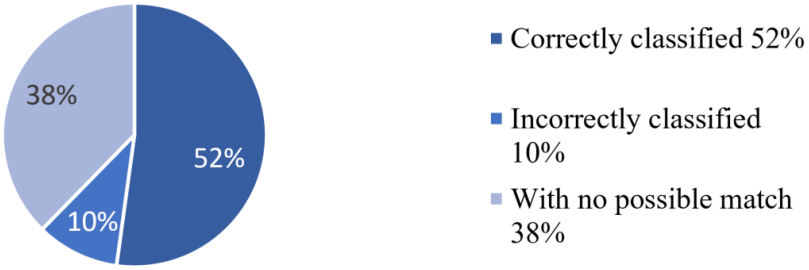

Figure 4.8: Distribution of best matches using Heuristic 2

It is a significant improvement over the first approach of the entity only matching, Section 4.7.2.1. The problem is now is how to target the messages with no possible correct matches to classify them as not related to news?

4.7.2.3 No possible matches

While there are no possible correct matches for 26 out of the 69 messages, as demonstrated at the end of Section 4.7, this is still a problem for the system, as it worsens the precision of the matching mechanism and inherently the accuracy of the labeling system.

To understand what type of matches are occurring with these messages, a simple analysis to the best matches shows they are characterized by having a low number of hits (largely 1 hit of keywords and/or 1 hit of entities).

In order to overcome the high number of these messages that have no possible correct match, a filtering system was developed. This filtering condition takes into account the number of hits of entities (e) and the number of hits of keywords (k) as seen in Table 4.4.

The first two conditions eliminate the messages with the lowest hits (weak matches). The second subset eliminates by the sum of hits. The third subset of conditions gives double weight to the hit of keywords due to its impact in separating correct from incorrect candidate matches (Figure 4.7).

The %Correct column is the percentage of correct matches from the best matches set. The %Lost column represents the percentage of correct messages that are lost when using the filtering condition (compared with no condition).

The filter condition chosen by its highest performance and heuristic was:

Table 4.4: Results of the condition filter using the hits of entities (e) and keywords (k)

Condition	Correct	Incorrect with match	Incorrect with impossible match	% Correct	% Lost
No condition	36	7	26	52,2%	0,0%
e>1 | k>1	36	7	8	70,6%	0,0%
e>1 & k>1	28	2	4	82,4%	22,2%
k + e > 3	34	4	4	81,0%	5,6%
k + e > 4	26	1	3	86,7%	27,8%
2k + e > 3	36	7	8	70,6%	0,0%
2k + e > 4	35	5	6	76,1%	2,8%
2k + e > 5	33	3	4	82,5%	8,3%
2k + e > 6	29	2	3	85,3%	19,4%

$$Filter\ condition: \ hits\ of\ entities + hits\ of\ keywords > 3 \tag{4.4}$$

The new *Heuristic* 3 will now use the highest hit of keywords (*Heuristic* 2) in conjunction with the new filtering condition:

$$Heuristic\ 3: hit\ of\ keywords + filter\ condition. \tag{4.5}$$

Applying now *Heuristic* 3 to the labeled dataset, this heuristic seems to eliminate most of the messages with no possible match, and in the process, a big part of incorrect matches, Table 4.5.

Table 4.5: Results using 3 different heuristics

	Correct	Incorrect with match	Impossible correct matches
Truth: labeled data	43	0	26
Heuristic 1	28	15	26
Heuristic 2	36	7	26
Heuristic 3	34	4	4

From the 43 originally labeled correct best matches, only 34 were classified as correct and passed through the filter using the heuristic with the best performance, *Heuristic* 3. Translated in performance, the system can correctly identify 79% of the correct matches and the matches labeled as correct have an accuracy of 81%.

4.7.2.4 Conclusions of all the approaches regarding the candidate matching experiment

- **Matching with entities:** the entities role was studied, as well as how it impacted the matching of news and posts (*Heuristic* 1). Although there seems to be some association between

the high number of entity hits and correctness of the candidate match, its correlation cannot be taken as granted.

- **Matching with keywords:** the role of keyword hits was analyzed. This feature appeared to better split incorrectly from correct candidate matches when compared to the previous approach. Using it as the main heuristic, *Heuristic* 2, for selecting the best match improved the results.

- **No possible matches:** the focus was on how to remove messages with no possible correct match from the best matching set. A filter condition was envisioned to eliminate these message, using the number of keywords and entities hits, *Heuristic* 3.

  Because the number of objects to study was relatively low (69 best matches), the filter will be tested in a bigger dataset.

4.7.3 Best matching experiment

In the previous experiment, it was stressed the ability of the system to correctly intersect analogous information between messages (published by news organizations) and news: candidate matches.

It is important to study this method with a bigger and broader range of messages (not coming from news organizations) and improve the overall performance of the system.

For this experiment, the system collected messages from news organizations (Facebook and Twitter), messages extracted from the trending topics on Twitter and news articles posted from t 8th to the 14th of April of 2017 (7 days), Table 4.6.

Table 4.6: Heatmap of the data collected by each dataset

Day of the month	8th	9th	10th	11th	12th	13th	14th
News	880	990	1642	1682	1934	1895	1577
SNRSS_FB	455	491	596	621	625	645	517
SNRSS_TW	932	954	1399	1402	1453	1492	1193
SNN_FB	442	388	519	581	631	577	494
SNN_TW	900	847	1346	1485	1715	1662	1455
SNT	99310	106064	158604	183241	180344	189602	176907

An interesting fact to note is that the number of news articles published on the 8th and 9th of April is significantly lower. This is probably related to the day of the week of those days (Saturday and Sunday), meaning that there are fewer published articles in the weekend. This same phenomenon is also visible in the number of messages shared on Facebook and Twitter (SNRSS, SNN, SNT).

The information extraction pipeline was then applied to the collected data and used to match each message with the article with the highest hits of keywords between them, *Heuristic* 2, Figure 4.9.

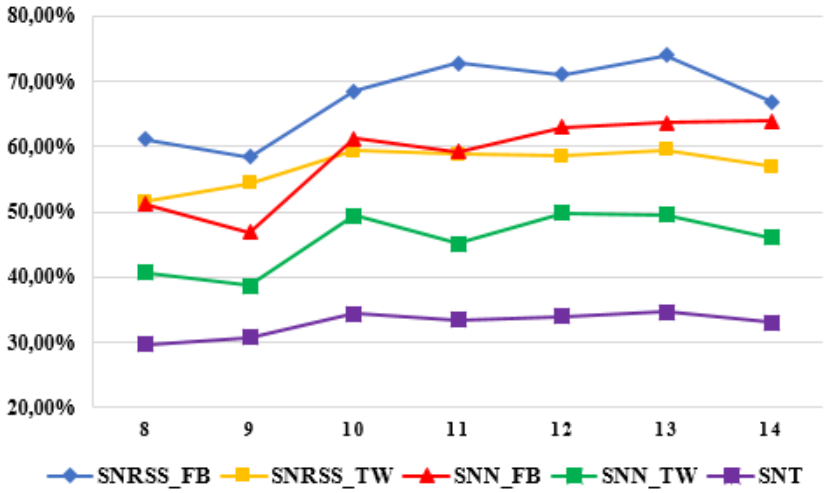

Figure 4.9: Percentage of messages matched at each day

Considering the results of the previous experiment, in Section 4.7.2.2, the accuracy of this matching system alone is not able to handle messages that most likely will not have a correct match. To overcome this downside, a filter condition (*Heuristic* 3) was proposed and tested in a small dataset, in Section 4.7.2.3, with promising results. To test this filter, a subset of the best matches was selected and manually labeled as correct or incorrect.

The subset was comprised of 10% of the messages of each day of the SNRSS and SNN and 1% of the SNT (due to the extremely high number of messages of this dataset), resulting in 5037 labeled matches, Figure 4.7.

Table 4.7: Labeled data

	TRUE		FALSE	
SNRSS_FB	160	60,6%	104	39,4%
SNRSS_TW	354	70,2%	150	29,8%
SNN_FB	98	44,1%	124	55,9%
SNN_TW	203	46,8%	231	53,2%
SNT	1117	30,8%	2507	69,2%

4.7.3.1 Filter condition

Applying now the filter condition (*Heuristic* 3), presented in Section 4.7.2.3, to the labeled data, the accuracy is of 84% and the confusion matrix presented in Table 4.8.

The filtering condition (*Heuristic* 3) does, in fact, produce results with a higher accuracy than the simple best matching with keywords (*Heuristic* 2). However, the purpose of the research is to develop a system with the highest accuracy possible. Because we are dealing with 5037 entries, it is not plausible to look at each result and try to come up with the best filter to detect false matches, manually.

Table 4.8: Confusion matrix using the filter condition

	True	False	Precision
True	1512	224	88.10%
False	411	2890	87.55%
Recall	78.63%	92.81%	

4.7.3.2 Filter model

To make sense of all the labeled data and create the best filter possible, this approach will use a machine learning algorithms to classify the matches as correct or incorrect, that is, a binary classification problem. This new heuristic will be:

$$Heuristic\ 4 : hit\ of\ keywords + filter\ model. \tag{4.6}$$

The first step to solving this problem is to reserve a subset of the labeled matches to only be tested at the end of all the improvements (holdout), preventing over-fit. This subset will be comprised of 10% of the data. The 90% of the remaining matches will be the focus of the training and testing machine learning models, to find the one with the best performance.

The first approach was to use the features already created: hits of persons, hits of locations, hits of organizations, hits of dates, hits of keywords, sum of hits of entities.

A feature previously used in the filtering condition, in Section 4.7.2.3, was the product of the hits of keywords with the hits of entities. This new value, called the score, will be used alongside the other features.

$$score = hits\ of\ entities \times hits\ of\ keywords \tag{4.7}$$

Machine learning models implemented in the R caret package [fJWWW+12] were trained, using 10-fold cross-validation, and applied to the testing set (results in Table 4.9).

Table 4.9: Performance using 7 features

	AUC	Accuracy	F-measure
SVM	94,2%	88,4%	85,0%
Naive Bayes	94,0%	88,2%	85,5%
Random Forest	94,1%	88,1%	85,2%
Neural Networks	94,9%	89,1%	85,8%
Decision Tree	90,0%	88,1%	85,7%

The results obtained by the machine learning models have already a higher accuracy (88.4%) than the one obtained by the filter (84% in Section 4.7.3.1).

Besides accuracy, AUC and F-measures, these models provide insight on the importance of features. Calculating the importance of the features (from 0 to 100), in Figure 4.10 , the highest-ranking ones are the hits of keywords and the score (the most relevant to train and test the models).

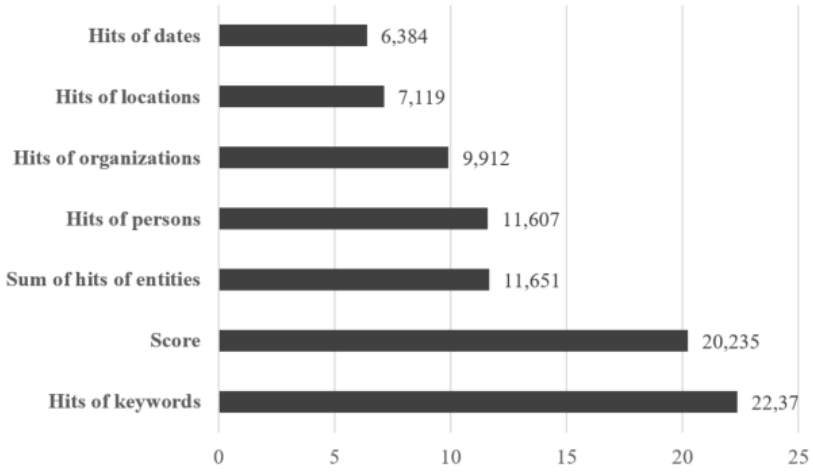

Figure 4.10: Importance of the 7 features using the random forest model

As previously stated in Section 4.7.2.2, the hits of keywords allow the system to choose the best candidate match for each pair of message and news. A fallback of this feature is that it does not consider the number of keywords in an article nor the length of a message.

Using the cosine similarity between the text of the message and the vector of keywords of a news article, the system has access to a new feature that considers the size of these two objects which could possibly improve the accuracy of the machine learning models.

These are some examples of the application of cosine similarity with values ranging from 0 to 1, using sentences of different length:

- **Sentence:** this is a test for cosine similarity with a big sentence

  Keywords: cosine

  Keywords similarity: 0.277

- **Sentence:** this is a test for cosine similarity with a big sentence

  Keywords: cosine, sentence

  Keywords similarity: 0.392

- **Sentence:** cosine similarity sentence

  Keywords: cosine, sentence

  Keywords similarity: 0.816

In the first and second example, a sentence of the same length is presented, where the second example has a higher hit of keywords (and consequently a higher keyword similarity). In the

second and third example, sentences differ in length but have the same hit of keywords. Here, the example with the smaller sentence has a higher keyword similarity.

Hence, the similarity of keywords feature was added to the data, the models retrained and tested and feature importance recalculated.

Table 4.10: Performance using 8 features

	AUC	Accuracy	F-measure
SVM	97,2%	92,3%	89,6%
Naive Bayes	96,6%	91,5%	88,3%
Random Forest	96,5%	91,0%	87,6%
Neural Networks	97,1%	92,4%	90,7%
Decision Tree	93,0%	90,5%	89,4%

With the added feature, the similarity of keywords is ranked the most important value, as seen in Figure 4.11, meaning that its addition brought performance improvements to the models, Table 4.10.

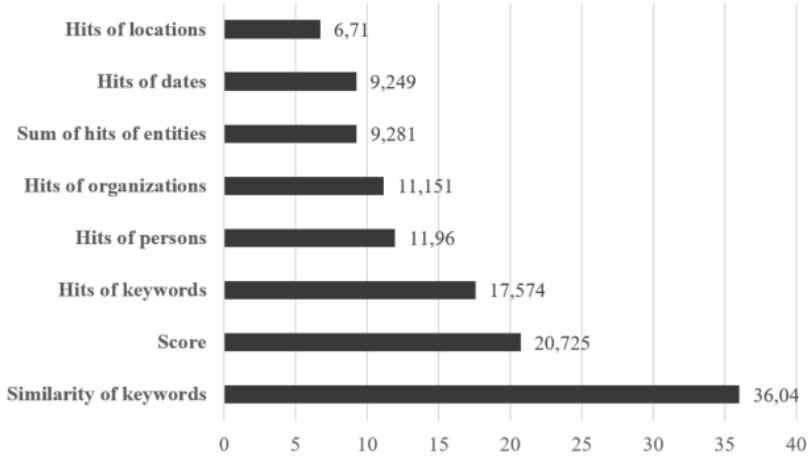

Figure 4.11: Importance of the 8 features using the random forest model

The models that had the highest performance in terms of accuracy, AUC and F-measure were SVM and Neural Networks, Table 4.10. Using the holdout dataset (10%) on these models, the results are similar, Table 4.11, meaning that there was no over-fit to the data.

Table 4.11: Performance using 8 features on the holdout subset

	AUC	Accuracy	F-measure
SVM	97,9%	93,3%	89,6%
Neural Networks	97,9%	93,9%	91,5%

Applying one of the trained models, in this case, SVM, to filter the best matches along the 7 days, Figure 4.12 , it is visible a significant decrease in the percentage of considered news related messages. Although there are fewer matches, the confidence in the whole system is higher.

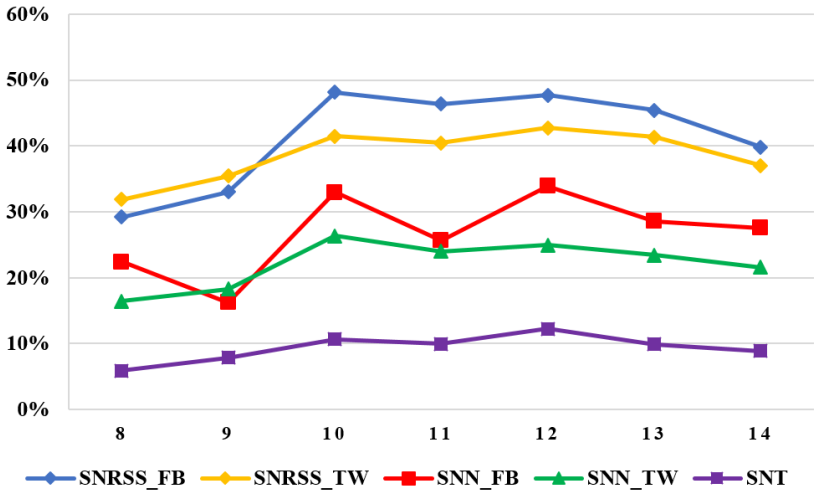

Figure 4.12: Percentage of messages matched at each day, with the filtering model

The datasets less affected by the filtering model were the ones coming from news organizations (SNRSS and SNN), Table 4.12, corroborating the assumption made in the beginning of the research, Section 4.5.2: messages published by news organizations would have a higher percentage of news related messages than messages from non-news organization (SNT), Figure 4.12.

Because the SNT dataset is the collection of messages posted by random users, a small percentage of news-related posts is expected, even when collecting posts with trending topics (reflected in Figure 4.12).

Table 4.12: Percentage of removed matches by the filtering model

Day of the month	8th	9th	10th	11th	12th	13th	14th
SNRSS_FB	52,16%	43,55%	29,66%	36,28%	32,88%	38,57%	40,46%
SNRSS_TW	38,25%	34,87%	30,12%	31,19%	26,94%	30,44%	34,90%
SNN_FB	56,19%	65,38%	46,23%	56,69%	46,10%	55,04%	56,96%
SNN_TW	59,56%	52,60%	46,70%	46,79%	49,94%	52,73%	53,20%
SNT	80,12%	74,49%	69,04%	70,18%	63,87%	71,49%	73,10%

4.7.3.3 Conclusions of the best matching experiment (7 days)

- **Filter condition:** the filtering condition was tested in a large labeled dataset, with the best matches, and proved to be an effective method to divide correctly from incorrect matches. However, a more precise method was needed to improve the performance of the system.

- **Filter model:** Machine learning algorithms were applied to the labeled data (with added features) to come up with a filtering model capable of classifying matches, as correct or incorrect, with improved precision over the filtering condition and overall system.

4.7.4 Matching method chosen

From *Assumption* 1, the definition of news relevancy was that: if a message has information present in a news article, then it has some degree of relevancy.

In Section 4.7.2, it is explored how to match messages with news, that is texts that share information. In Section 4.7.3, the selection of the best match was studied and its performance improved.

After evaluating several approaches to the candidate and best matching between news and messages, the chosen heuristic was *Heuristic* 4. This heuristic allows the system to select the best candidate match, using the highest hits of keywords and to filter the weakest matches using the filtering model trained in Section 4.7.3.2.

The final accuracy of the matching method is of 92% in selecting the correct best matches and eliminating the weak ones. If *Assumption* 1 proves to be true, this means that the accuracy of the labeling system is also 92%.

4.7.5 News topics

Besides studying the rate of newsworthy messages regarding their social network source, it is also interesting to analyze which topics of news articles are more relevant. As presented in Section 4.5.1, topics are obtained by the divisions of news pieces that news organizations provide through different RSS feeds with specific themes and topics.

For the study of the impact of news topics, the dataset presented in Section 4.7.3 will be used alongside with the matching heuristic chosen in Section 4.7.4.

The following analysis is based on data presented in the Appendix A, in Table A.2 and Table A.3.

Analyzing the distribution of news articles' topics, there is a pattern that separates weekend from weekdays: in the weekend, the topics with more collected articles are coming from world and sports RSS feeds. Whilst, sports account for 30% of the collect news pieces, in weekdays, this percentage drops to 20% or less. During the weekdays, the topic distribution of news pieces is more even across areas.

The focus of the analysis will now shift from the distribution of collected news articles to the distribution of news pieces matched with newsworthy messages. The purpose of this analysis is not only to find the topic with most newsworthy messages but to understand if there are differences of discussed topics depending on the source or social network.

Studying the distribution of the most discussed topics by each day and dataset, an interesting fact to note is the relevance that sports have in Twitter trending topics. During the weekends, sports is the most discussed newsworthy topic in this network. During the weekdays, the presence

of sports does not correspond with the number of collected news articles of this topic (third most collected articles but the second most talked topic), leading to the conclusion that sports play a big role on Twitter. These results corroborate the scientific research that has been conduct in studying the role of sports in the social network Twitter [KLPM10].

When the most discussed topic is not sports, news articles coming from the topic world take the first place in the number of matched news with newsworthy messages. This is to expect as this topic is the one with most collected articles.

4.8 Message classification

Looking back at Section 4.2, the labeling of social network messages usually relied on the manual classification of these texts. The manual effort, as previously stated, presents great hindrances to the development and expansion of any project due to its investment and precision costs.

In the final stage of the built system, social network messages are labeled automatically as newsworthy or not. This method presents itself as a substitute to the manual labeling and to provide researchers with a powerful tool capable of supporting the research in the field of social networks.

4.8.1 Labeling

The built system to label messages relies on *Assumption* 1 presented in Section 4.2.2. In this conjecture, a message is considered to have a degree of news relevancy if its information, or part of it, is present in a news article. Therefore, a matching system was developed to emulate this intersection of information.

In this system, the degree of shared information was calculated by how similar certain features of both texts were. As presented in Section 4.7.2, features such as entities and keywords were used to detect messages and news articles that potentially shared the same information: candidate matches.

If a social network message is not matched with any of the news articles collected by the system, then, by *Assumption* 1, the message is classified as not newsworthy. On the other hand, if there is shared information (candidate matches), the message was classified as newsworthy.

As pointed out in Section 4.7.2.3, even when selecting the best match over the candidate matches, these might still be incorrect. To address this issue, a filter model was developed to filter out weak matches and classified them as not newsworthy, in Figure 4.13. The remaining matches are classified as newsworthy.

The filter model developed can be taken upon as an advantage or disadvantage to the automatic labeling system. On one hand, the system provides assurance that messages are only classified as newsworthy with a high degree of certainty. On the other, the system might be incorrectly classifying newsworthy messages as not newsworthy. This is an interesting point to go over: sensitivity over specificity.

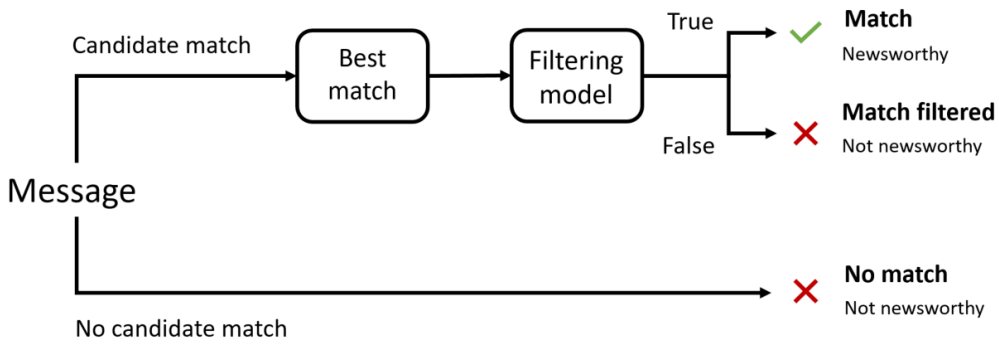

Figure 4.13: Message classification scheme

Since the ultimate purpose of the system is to evaluate the accuracy of a classification model, the underlying question between sensitivity and specificity is the following: should the system give room for errors on the labeling of newsworthiness (sensitivity) or not (specificity)? The answer to this question falls back to the priority of the research.

To the project REMINDS, it is worst to identify a not newsworthy message as a newsworthy than the other way around, hence, the precision of the system is prioritized.

By using the filtering model, the system is only labeling as newsworthy the messages with a high degree of certainty, that is, the system has a high specificity (True Negative Rate).

The architecture of the labeling system was composed and built around *Assumption* 1. This means that if a message has the same content as a news article, they share similar or the same information, so the system should label them as newsworthy. On the other hand, we cannot assert with certainty that a message is not newsworthy if there are no matches between messages and news articles. While in this instance, the system will be labeling them as not newsworthy, the only assurance we have is that their content is not present in the collected articles.

In the same logic, messages filtered out by the filter model should be taken into consideration, since, by the system, the shared content is not enough to be considered a relevant match, although that does not rule out the possibility of the message being newsworthy: only that the system did not collect any news article related to the news event.

4.8.2 Discussion about labeling validation

Assumption 1 has been taken as granted along the development of the labeling system, serving as the underlining basis for the classification of social network messages. The evaluation and validation of the system are needed to assert whether the assumption proves to be correct or not.

Until this point, experiments regarding the built system have been focused on the accuracy of matching news with messages, not considering the newsworthiness of the actual message. In the next experiment, the ability of the system to label correctly newsworthy messages will be focused.

To validate the developed system and obtain an accuracy regarding the actual classification of messages as newsworthy or not, an experiment was conducted on the previously collected data, in Section 4.7.3. The outline process of this experiment was the following: to use the developed system to label automatically messages as newsworthy or not and then to assert its precision using manual labeling.

As seen in Figure 4.13, the process to label messages on their newsworthiness can have three distinct paths in the system. The three manually labeled datasets were the following

1. The first data set to be evaluated was composed of messages that presented candidate matches (messages matched with collected news articles) and that the filtering model deemed not weak (the probability of the match being correct is high), being labeled as newsworthy.

2. The second set of messages will be comprised of messages that although had candidate matches with news pieces, were filtered out by the model, and therefore labeled as not newsworthy.

3. The third path, and its consequent dataset, to be evaluated was composed of messages that had no candidate matches, meaning that there were no news articles that shared information with these messages, ruling them as not newsworthy.

For this experiment, we collected 10 messages of each day, and run through the automatic classification system, resulting in 1050 messages labeled by the system. This resulting number is explained by the following equation:

$$Number\ of\ messages = 10(messages) \times 7(days) \times 5(datasets) = 1050 \tag{4.8}$$

Each message was then manually labeled on their newsworthiness. With this manual work, the precision of each path of the automatic labeling system can be evaluated. The obtained results are presented in Table 4.13 and in more detail in Appendix A , Table A.4.

Table 4.13: Performance evaluation on the three labeling paths of the system

	Social Network	**Match**	**Match filtered**	**No match**
SNRSS	FB	90.0%	64.3%	90.0%
	TW	94.3%	50.0%	67.1%
SNN	FB	80.0%	58.6%	82.9%
	TW	80.0%	64.3%	90.0%
SNT	TW	72.9%	75.7%	97.1%
Average		83.1%	62.3%	83.1%

The columns of Table 4.13 represent the following:

- **Match:** Precision metric of the class newsworthy in labeling messages that presented matches and not ruled out by the filtering model.

- **Match filtered:** Precision metric of the class not newsworthy in labeling messages that presented matches, but that were filtered out by the model.

- **No match:** Precision metric of the class not newsworthy in labeling messages that did not matched any of the collected news articles.

As it would be expected, the precision of the system in labeling newsworthy messages was higher in the dataset with the highest number of news-related messages (Social Network RSS feed dataset). On the other hand, the precision of the system took a hit when labeling newsworthy messages on the dataset with the lowest newsworthy probability (Social Network Trending dataset).

The precision of the system in labeling not newsworthy messages was significantly higher when there were no candidate matches to the messages (No match). Taking into account *Assumption* 1, this is an expected result: if there is no shared information between a message and any news article, then this message has no news relevancy.

Both 'Match' and 'No match' are the opposite extremes of the based assumption: either the message has overlapping information and is not filtered out by the model, or there is no overlapping information between the message and news pieces. This means that the accuracy of these paths of the labeling system would get the highest precision (both with a precision of 83.1%).

The more dubious area of the automatic labeling system are messages that had one or more candidate matches but were considered, by the filtering system, to be weak matches. As seen in Table 4.13, this precision was the lowest across all paths of the system.

As discussed in Section 4.8.2, the precision of the labeling of newsworthy messages was chosen over the recall, meaning that, for the classification model, the highest priority is to have the highest specificity in the classification of newsworthy messages. A high specificity allows the system to be resilient to false alarms, or false newsworthy messages.

$$specificity = \frac{TN}{TN+FP} = \frac{518}{518+59} = 89,8\%$$

$$sensitivity/recall = \frac{TP}{TP+FN} = \frac{291}{291+182} = 61,5\%$$

$$precision = \frac{TP}{TP+FP} = \frac{291}{291+59} = 83,1\%$$

The accuracy of the system is of 77% and F-measure 70.7%. Regarding these results, we should take into account the small dataset labeled, the number of news organizations used as source (13) and the overall unreliability of the tools being used in the labeling system.

More detailed information regarding the results of this experiment can be found in Appendix A, Table A.5.

One approach to increase the precision and recall of the labeling system might be to add more news sources to the labeling system. Keeping in mind that, for these experiments, only 13 organizations were used as sources of the news knowledge base, these can be considered a satisfactory result. It is to expect that the accuracy of the method would increase with the addition of new news sources. As previously referenced, in Section 4.5, the system was built in a way that to add new news sources, the researcher only needs to add the RSS feeds links to a list.

While the presented experiment showed promising results for labeling messages, we should acknowledge the scope of this test. The system only collected news from 13 organizations, meaning that the knowledge base of the labeling tool is bound to the publications of these entities. While we might add new sources to the news aggregation system, a fact still remains: the system will never collect enough articles to correctly label all social network messages.

Therefore, the confidence in the label "newsworthy" will be higher than the confidence in labeling "not newsworthy". This is due to the problem of our finite knowledge base:

- If the system matches a message with a news article, we have reason to believe that the information present in the message is also present in the article, and by *Assumption* 1, it is newsworthy. This evaluation can be checked by comparing the content of both message and article.

- If the system does not find any, or relevant, intersection between a message and an article, we cannot be sure that there is no article published regarding the content or event referenced, because the system does not collect all of them.

4.9 Conclusions

The purpose of the presented work was to develop an automated system capable of assessing the accuracy of a relevance classification model in social networks with some degree of certainty.

In Section 4.2, the current state of the research field in social networks was explored. The need for labeled data was presented: to measure the precision of the models and work being developed. The evaluation of any system is what allows any research team to develop and improve their system. Without evaluation, researchers would not have any means to compare the several approaches to the problem and assess if their work was improving the research.

Although this dissertation was developed under a research project, the presented work was developed to be fully independent. For this reason, the data aggregation consisted of both news articles and social network messages, in Section 4.5. After the data collection, to compare both types of texts, an information pipeline was applied to the both data sources: information extraction, in Section 4.6.

After extracting the data and organizing the information, the system was now capable of comparing information coming from social networks and news organizations. In Section 4.7, the

notion of matching messages and news was explored and several experiments were conducted to validate this method of comparing information.

The final stage of the built system was the labeling of messages as newsworthy or not, in Section 4.8. In this stage, the overall performance of the system was tested, stressing the assumption made at the beginning of this work: the assumption that messages sharing similar information with news will have a agree of relevancy.

In the end, the proposed and built system proved to be able to correctly label messages as newsworthy with high accuracy from both news and non-news organizations, with a precision of over 80 % in newsworthy labeling.

Chapter 5

Conclusions

In Section 5.1 we present the conclusions that can be drawn from the developed project and in Section 5.2 we discuss some of the frailties and weaknesses of the implemented work. The contributions to the scientific community and applications of this project are presented in Section 5.3 and Section 5.4 respectively. In Section 5.5, we suggest some improvements to this work.

5.1 Synthesis

In this dissertation, we presented a labeling tool capable of classifying messages on their news relevancy. Its purpose is to provide researchers working with social network data to assess the precision of their work automatically, without the intervention of human classification.

The problem of labeling data in social networks was explored: the monetary, temporal, and precision costs of this exercise might rule out continuous improvement and development of the research. With this problem in mind, the advantage that the proposed work was going to bring to the scientific community was clear: an automatic mechanism that, with some degree of precision, allows the accuracy assessment of a model with no added cost and up-to-date data.

The proposed and built solution allows researchers to obtain the labeling of social network data relative to their newsworthiness with no human interaction. The developed labeling tool is an implementation of the assumption: if a message shares information with a news article, it should have a degree of relevancy. Hence, this project relies heavily on the creation of a knowledge base of news. The system depends on its knowledge base to label messages as newsworthy or not.

The labeling tool developed has four distinct phases: the aggregation of data, the extraction of information, the matching of news with messages and the classification of messages. The end goal is for this labeling tool to replace the role that human classification plays on research in social networks.

In the aggregation of data, the system collected news articles from specific list of trusted sources. The collection of news articles was made by the RSS feeds that these news organizations

provide. Whilst RSS feeds were provided with some of the information of the articles, important data such as the text of the article was still missing. A Python library was used to perform web scrapping techniques on each of the collected URLS from the RSS feeds to provide with a more complete set of attributes and information about each news piece.

Since the developed system is to be embedded in project REMINDS, the collection of social network messages was not a necessary step for this tool. However, we intended to create and automatous mechanism capable of being used by any research project or application. Hence, a social network messages aggregation system was also built. This system collected messages published by news and non-news organizations from social networks Twitter and Facebook using their APIs.

To compare the information between messages and news, an information extraction method was added to the system. This extraction was implemented in a way that the same information could be extracted from both messages and news. Named entity recognition tools where applied to the texts and entities were extracted and stored.

The matching system provides a mechanism to compare information present in messages and news. The evaluation of shared information is based on how similar the extracted information is between texts. If a message and a news article share the slightest information, they are considered a candidate match. Several heuristics were tested to choose the best candidate matches and to rule out weak candidate matches. The best heuristic opts by choosing the best candidate match based on keywords (how many times a news article's keyword appears in a message's text) and in a filtering model (classification algorithm created by labeled candidate matches) to filter out weak matches. The matching system was validated along its implementation, and in the end, the accuracy of the filtering model was 92.3% using an SVM classifier.

The labeling of messages used the matched messages, labeling them as newsworthy, and messages filtered out by the filter model and messages with no candidate matches were labeled as not newsworthy. Evaluating the system, its precision in labeling newsworthy messages was higher for messages coming from news organizations. On the other hand, not newsworthy classification had a higher precision in messages not coming from news organizations.

The evaluation made to the labeling tool resulted in a newsworthy precision of 83.1% and a not newsworthy precision of 74.0%.

This validation leads us to conclude that the system can in fact be used by REMINDS and other research projects to label their study data and assess the accuracy of the system.

5.2 System discussion

One of the main weakness of the presented project is the fact that relies in several technologies and tools, each one with a degree of unreliability to execute the intended work:

- In the collection of news data, the system relies in a Python library to extract all the information possible from a web-site using web scrapping scripts.

While the system would usually collect correct information, such as publication date and artcile's text from the URL of an article, there were some sites in which the system failed to extract correctly the information, and had to be removed from an initial RSS list. This might be explained by the fact that those sites did not follow any standard document structure for news content, debilitating the ability of the library to extract the relevant information.

Therefore, while the system can extract news information from URLs, it restricts the organizations used by the news aggregator to those that are compatible with the library.

- Named entity recognition, used in the information extraction pipeline of the developed project, has a varying percentage of accuracy on identifying entities in texts coming from the news wire and texts coming from social networks.

  Analyzing the results that NER tools can have in different types of text might explain some of the outcomes of this system. Looking at the stufy from Bontcheva et al [BDF+], where the performances of Stanford CoreNLP were stressed with different types of texts, it is visible a clear discrepancy between news texts and social network texts. The F-1 performance measure for NER in news articles is of 89%, while in texts coming from social networks, such as Twitter, is 41%. This radical difference between performance measures, on the texts subject to our project, can influence the outcome of the labeling tool developed and its precision, as entities are one of the main features of this system.

  The low reliability of NER in social network messages might be pointed out as one of the reasons for the poor performance of the matching mechanism only using entities, as presented in Section 4.7.2.1.

Besides the unreliability of the tools used in the development of this project, the knowledge based in which the system basis its decision on could lead to correct or incorrect labeling of messages.

The labeling tool classifies a social network message as newsworthy if its content is present in the database of news articles collected. This means that, for the labeling tool, being present in a news article is a synonym to being newsworthy.

As stated before in Section 4.8.2, while the system tries to make the most of the collected information, it will never aggregate all the articles published by every news organization and topic of public interest. Therefore, the labeling tool will never correctly classify all the collected messages.

Thus, the classification as a newsworthy message is more precise, as it is a concrete relation between a message and a collected article, than a classification as not newsworthy message. Labeling a messages as not newsworthy only means that the system did not collect an article related to the message, not ruling out the possibility that the message could be, in fact, newsworthy).

5.3 Contributions

The contributions of this project to the scientific and research community are the following:

- The study of social network messages from different social networks (Facebook and Twitter) and from different sources (news and non-news entities) from the perspective of newsworthiness of their content.

- The study of content similarity between texts of different sources (news wire and social networks), different levels of complexity (formal and informal text) and different lengths (articles and short messages).

- The assessment of the impact that entities and keywords have when comparing the content of messages and news

- The development of a matching system, to pair a message with the most similar news article. This matching mechanism is based on the hits of features between both texts (number of feature values present in both messages and news). A filtering model was also developed to eliminate irrelevant matches (weak matches) between messages and news.

- The development of an accuracy assessment system that updates itself, continuously, with new information from the news wire and social networks, independently from the project.

5.4 Work applications

While the presented system proved to be successful for automatic assessment of accuracy in a classification model, the developed mechanism of matching social network messages and news articles can prove to be useful in other areas and applications.

A subject of increasing interest from the scientific community is the detection of fake news in social networks. As stated in Section 4.5.1, measures were applied to prevent the collection of news pieces from untrusted sources. An application of the system could be to use it as a fake news detection system in social network messages. This process would be used to not only find news worthy messages, as well as messages with notorious news related features that reference contradictory information to the one present in news articles.

The same process of detecting fake news could also be used by news organizations to find contradictory information. This method has the potential to help news entities find information shared in social networks that are still not present in news articles or inconsistent with them. This is an important aspect, considering the role that the citizen journalist can have in sharing information of what is happening around them. News organization could use this information to investigate the new data, contact the users and update their stories as live information comes through.

5.5 Future work

Regarding the presented system, there is some work that can be done to further enhance it.

The newsworthiness of a social network message relies both in its content and in its well-timed publication. The matching mechanism developed restricts the pairing of messages with news published in the same day. One of the improvements that could be applied to the system is two folded: relax the time frame of the matching mechanism, more than one day, and consider the difference in publishing time between the message and the news. With this improvement, the system could match messages with news that were published before or afterwards. With the publishing times' discrepancy, the system could assign a higher probability of relevancy to messages and news published at the same time. If the message was published before the news, this relevancy could be considered even higher.

Another area that could be explored is the frequency to which entities are present in social network messages and news articles. As previously stated, some common entities can add noise to the matching system due to their high recurrence. The frequency of entities could be studied to associate a value of relevancy confidence [SB88, AHSW11]. High frequent entities would be associated with a lower value of relevancy [JON72]. With this method, the system could handle the noise introduced by recurrent entities and improve the overall performance of the matching mechanism.

One subject that failed to be included in the development of the system is event detection.

While there are several sources of news being used to collect news articles, this information is not being enhanced by the system, meaning that the evaluation of information is done each article at a time. A promising approach would be to try to make sense of the information present in the different news pieces and come up with a base of knowledge for each news or event being discussed in the articles [SHM09, ZZW07].

The application of event detection can provide with the clustering of news articles [MBE+02, RBGZR01]. With this method, the system could associate different articles to the same news. Information coming from different sources about the same topic and event would be assembled and potentially provide with an additional matching mechanism of news and messages, improving the performance of the system [BNG11a, PSPY13].

References

[AG04] Eugene Agichtein and Venkatesh Ganti. Mining Reference Tables for Automatic Text Segmentation. *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 20–29, 2004.

[AHSW11] Sitaram Asur, Bernardo A. Huberman, Gabor Szabo, and Chunyan Wang. Trends in Social Media: Persistence and Decay. *SSRN Electronic Journal*, feb 2011.

[AVSS12] Puneet Agarwal, Rajgopal Vaithiyanathan, Saurabh Sharma, and Gautam Shroff. Catching the Long-Tail: Extracting Local News Events from Twitter. *Sixth International AAAI Conference on Weblogs and Social Media*, pages 379–382, 2012.

[BDF+] Kalina Bontcheva, Leon Derczynski, Adam Funk, Mark A Greenwood, Diana Maynard, and Niraj Aswani. TwitIE: An Open-Source Information Extraction Pipeline for Microblog Text.

[BGS06] Alexander Blekas, John Garofalakis, and Vasilios Stefanis. Use of RSS feeds for content adaptation in mobile web browsing. *Proceedings of the 2006 international cross-disciplinary workshop on Web accessibility (W4A) Building the mobile web: rediscovering accessibility? - W4A*, (May):79, 2006.

[BHBL09] Christian Bizer, Tom Heath, and T Berners-Lee. Linked data-the story so far. 2009.

[BHIBL] Christian Bizer, Tom Heath, Kingsley Idehen, and Tim Berners-Lee. Linked Data on the Web (LDOW2008).

[BMSW97] Daniel M. Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. Nymble: a high-performance learning name-finder. *Proceedings of the fifth conference on Applied natural language processing* -, pages 194–201, 1997.

[BNG11a] Hila Becker, Mor Naaman, and Luis Gravano. Beyond trending topics: Real-world event identification on Twitter. *Icwsm*, 11:1–17, 2011.

[BNG11b] Hila Becker, Mor Naaman, and Luis Gravano. Beyond trending topics: Real-world event identification on Twitter. *Icwsm*, pages 1–17, 2011.

[CK04] Jeremy J Carroll and Graham Klyne. Resource Description Framework ({RDF}): Concepts and Abstract Syntax. Technical report, W3C, 2004.

[CL96] J Cowie and W Lehnert. Information extraction. *Communications of the ACM*, 39(1):80–91, 1996.

REFERENCES

[CMP11] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 675–684. ACM, 2011.

[CMP13] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Predicting information credibility in time-sensitive social media. *Internet Research*, 23(5):560–588, 2013.

[Fac17] Facebook Graph APIs, REST API. Available at https://developers.facebook.com/docs/graph-api, January 2017.

[fJWWW+12] Max Kuhn. Contributions from Jed Wing, Steve Weston, Andre Williams, Chris Keefer, and Allan Engelhardt. *caret: Classification and Regression Training*, 2012. R package version 5.15-044.

[Fre95] James C French. What is metadata. In *Proceedings of the SDM–92 Workshop: The Role of Metadata in Managing Large Environmental Science Datasets*, pages 3–8. Pacific Northwest Laboratory, 1995.

[FSF16] Alvaro Figueira, Miguel Sandim, and Paula Fortuna. An Approach to Relevancy Detection: Contributions to the Automatic Detection of Relevance in Social Networks. pages 89–99. 2016.

[Hau10] Austin Haugen. *The Open Graph Protocol Design Decisions*, page 338. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[JG15] geoffjentry@gmail.com Jeff Gentry. Package twitteR, R Based Twitter Client. Available at https://CRAN.R-project.org/package=twitteR, July 2015.

[JON72] KAREN SPARCK JONES. A STATISTICAL INTERPRETATION OF TERM SPECIFICITY AND ITS APPLICATION IN RETRIEVAL. *Journal of Documentation*, 28(1):11–21, 1972.

[KLPM10] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is twitter, a social network or a news media? In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 591–600, New York, NY, USA, 2010. ACM.

[LCLZ] Janette Lehmann, Carlos Castillo, Mounia Lalmas, and Ethan Zuckerman. Transient News Crowds in Social Media.

[LDC10] Vasileios Lampos, Tijl De Bie, and Nello Cristianini. Flu Detector - Tracking Epidemics on Twitter. In *Machine Learning and Knowledge Discovery in Databases*, pages 599–602. 2010.

[MBE+02] Kathleen R. McKeown, Regina Barzilay, David Evans, Vasileios Hatzivassiloglou, Judith L. Klavans, Ani Nenkova, Carl Sable, Barry Schiffman, and Sergey Sigelman. Tracking and summarizing news on a daily basis with columbia's newsblaster. In *Proceedings of the Second International Conference on Human Language Technology Research*, HLT '02, pages 280–285, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

[McC05] Andrew McCallum. Information extraction: Distilling Structured Data from Unstructured Text. *Queue - Social Computing*, 3(9):48–57, 2005.

REFERENCES

[MJD07] Gurmeet Singh Manku, Arvind Jain, and Anish Das Sarma. Detecting Near-duplicates for Web Crawling. *Proceedings of the 16th International Conference on World Wide Web*, pages 141–150, 2007.

[MP15] Amy Mitchell and Dana Page. The Evolving Role of News on Twitter and Facebook. *of Journalism Research Dana Page Communications Manager*, 202419, 2015.

[MPSR01] Filippo Menczer, Gautam Pant, Padmini Srinivasan, and Miguel E Ruiz. Evaluating Topic-driven Web Crawlers. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, pages 241–249, New York, NY, USA, 2001. ACM.

[MRMN14] S Munzert, C Rubba, P Meißner, and D Nyhuis. *Automated Data Collection with R: A Practical Guide to Web Scraping and Text Mining*. Wiley, 2014.

[MSB+14] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J Bethard, and David McClosky. The {Stanford} {CoreNLP} Natural Language Processing Toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014.

[Mur11] Dhiraj Murthy. Twitter: Microphone for the masses? *Media, Culture & Society*, 33(5):779–789, 2011.

[New17] Newspaper, Python Library. Available at https://pypi.python.org/pypi/newspaper, January 2017.

[nut17] Apache Nutch. Available at http://nutch.apache.org/, January 2017.

[NYT17] The New York Times Developer Network, APIs Fit to POST. Available at https://developer.nytimes.com/, January 2017.

[PGFA17] Alexandre Pinto, Hugo Gonçalo Oliveira, Álvaro Figueira, and Ana Oliveira Alves. Predicting the Relevance of Social Media Posts Based on Linguistic Features and Journalistic Criteria. *New Generation Computing*, pages 1–22, apr 2017.

[PO13] S Petrovic and Miles Osborne. Can twitter replace newswire for breaking news. ... *AAAI Conference on* ..., 2011:713–716, 2013.

[POL10] Saša Petrović, Miles Osborne, and Victor Lavrenko. Streaming first story detection with application to twitter. *NAACL HLT 2010 - Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Proceedings of the Main Conference*, (June):181–189, 2010.

[PSPY13] T Poibeau, H Saggion, Jakub Piskorski, and Roman Yangarber. Multi-source, Multilingual Information Extraction and Summarization. *Theory and Applications of Natural Language Processing*, pages 23–50, 2013.

[R C12] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.

REFERENCES

[RBGZR01] Dragomir R. Radev, Sasha Blair-Goldensohn, Zhu Zhang, and Revathi Sundara Raghavan. Newsinessence: A system for domain-independent, real-time news clustering and multi-document summarization. In *Proceedings of the First International Conference on Human Language Technology Research*, HLT '01, pages 1–4, Stroudsburg, PA, USA, 2001. Association for Computational Linguistics.

[RGSL04] D C Reis, Paulo B Golgher, a S Silva, and a F Laender. Automatic web news extraction using tree edit distance. *Proceedings of the 13th conference on World Wide Web WWW 04*, page 502, 2004.

[RMEC12] Alan Ritter, Mausam, Oren Etzioni, and Sam Clark. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 1104–1112, New York, NY, USA, 2012. ACM.

[SB88] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523, 1988.

[SCM99] Stephen Soderland, Claire Cardie, and Raymond Mooney. Learning Information Extraction Rules for Semi-Structured and Free Text. *Machine Learning*, 34:233–272, 1999.

[SHM09] Hassan Sayyadi, Matthew Hurst, and Alexey Maykov. Event detection and tracking in social streams. In *Icwsm*, 2009.

[SOM10] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 851–860, New York, NY, USA, 2010. ACM.

[Sta17] Stanford Named Entity Recognizer, The Stanford Natural Language Processing Group. Available at http://nlp.stanford.edu:8080/ner/, January 2017.

[ŠTP+13] Tadej Štajner, Bart Thomee, Ana-Maria Popescu, Marco Pennacchiotti, and Alejandro Jaimes. Automatic Selection of Social Media Responses to News. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 50–58, New York, NY, USA, 2013. ACM.

[STS+09] Jagan Sankaranarayanan, Benjamin E Teitler, Hanan Samet, Michael D Lieberman, and Jon Sperling. TwitterStand : News in Tweets . In *ACM GIS 2009*, 2009.

[Twi17] Twitter APIs, REST API. Available at https://dev.twitter.com/rest/public, January 2017.

[Vie10] Sarah Vieweg. Microblogged contributions to the emergency arena: Discovery, interpretation and implications. *Computer Supported Collaborative Work*, pages 515–516, 2010.

[ZCH+14] Deyu Zhou, Liangyu Chen, Yulan He, Information Integration, and Applied Science. A Simple Bayesian Modelling Approach to Event Extraction from Twitter. *Acl*, pages 700–705, 2014.

REFERENCES

[ZZW07] Kuo Zhang, Juan Zi, and Li Gang Wu. New event detection based on indexing-tree and named entity. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 215–222, New York, NY, USA, 2007. ACM.

REFERENCES

Appendix A

A.1 Data collection

In Table A.1 the sources for both the news articles and social network aggregation is presented. SNRSS corresponds to the dataset constituted by shared messages of organizations whose articles where collected. SNN is the dataset of collected messages from news organizations whose articles were not collected. FB stands for Facebook and TW for Twitter.

A.2 Matching of messages with news

In Table A.2, the topics with which messages matched the most are presented. In each cell, the first and second most matched topic is presented. This topics are the ones used in the aggregation of news, that is, the topic of the RSS feed used. W stands for world, S for sports, E for entertainment, B for business, P for politics. In this table, it is shown the importance of Sports in Twitter.

In Table A.3, the percentage of collected news pieces for each day and topic are presented. In this table, it is shown the decline in collected sports articles between the weekend (8th and 9th) to the rest of the week days.

A.3 Message classification

In Table A.4, it is presented the results obtaind by evaluating manually the labeling tool. Match, Match filtered and No match are the paths that the tool can take to label messages as newsworthy or not. For each path and dataset, 70 messages were evaluated manually.

In the Match path, messages are labeled as newsworthy and PT corresponds to the precision of newsworthy labeling (T / 70). IN both Match filtered and No match, PF corresponds to the precision on labeling not newsworthy messages (F / 70).

In Table A.5, the confusion matrix is presented with the Precision and Recall of the labeling tool for the classification of newsworthy and not newsworthy messages.

Table A.1: Collected datasources

	News	SNRSS_FB	SNRSS_TW	SNN_FB	SNN_TW
ABC News	x	x	x		
BBC News	x	x	x		
CNN	x	x	x		
Daily Mail	x	x	x		
The Economist	x	x	x		
Forbes	x	x	x		
The New York Times	x	x	x		
Reuters	x	x	x		
Sky News	x	x	x		
The Telegraph	x	x	x		
The Guardian	x	x	x		
Washington Post	x	x	x		
The Wall Street Journal	x	x	x		
TED				x	x
Business Insider				x	x
Fox News				x	x
The Huffington Post				x	x
NBC News				x	x
USA TODAY				x	x
Mashable				x	x
New York Post				x	x
MTV News				x	x
Los Angeles Times				x	x
The Scientist				x	x
Fortune Magazine				x	x
Science daily					x
U.S. News and World Report				x	x
Daily Wire				x	x
One Green Planet				x	x
SparkPeople.com				x	x
Tech Viral				x	x
Young Entrepreneur				x	x
Backstage Magazine				x	x
Game Informer				x	x
Seeker Daily				x	
YouBeauty				x	
Outlook, and PostEverything				x	
Tampa Bay Times				x	x
NaturalNews.com				x	x
Harvard University				x	x
The University of Texas at Austin				x	x

Table A.2: Most matched topics by day

	8th	9th	10th	11th	12th	13th	14th
SNRSS_FB	W E	W E	W B	W P	W P	W B	W B
SNRSS_TW	W S	W S	W B	W B	W B	W B	W B
SNN_FB	W S	W S	W B	W B	W P	W E	W E
SNN_TW	W S	W S	W B	W B	W P	W B	W B
SNT	S W	S W	W S	W S	W S	W S	W S

Table A.3: Percentage of collected articles by day and topic

	8th	9th	10th	11th	12th	13th	14th
World	36,5%	34,3%	32,5%	32,5%	35,2%	30,9%	33,8%
Sports	32,8%	29,9%	17,5%	17,8%	17,1%	18,4%	21,6%
Entertainment	10,9%	12,0%	11,1%	10,5%	11,4%	14,5%	15,0%
Business	10,6%	13,5%	21,9%	19,7%	17,8%	18,9%	16,0%
Politics	4,8%	6,7%	5,2%	6,9%	7,7%	5,6%	4,7%
Techonology	3,8%	3,2%	7,1%	6,4%	6,4%	6,6%	5,2%
Health	0,6%	0,3%	4,6%	6,0%	4,1%	4,6%	3,6%
Science	0,1%	0,0%	0,2%	0,2%	0,3%	0,5%	0,1%

Table A.4: Precision evaluation of the labeling system.

		Match			**Match filtered**			**No match**		
		T	**F**	**PT**	**T**	**F**	**PF**	**T**	**F**	**PF**
SNRSS	**FB**	63	7	90.0%	25	45	64.3%	7	63	90.0%
	TW	66	4	94.3%	35	35	50.0%	23	47	67.1%
SNN	**FB**	56	14	80.0%	29	40	58.6%	12	50	82.9%
	TW	56	14	80.0%	25	45	64.3%	7	63	90.0%
SNT	**TW**	50	20	72.9%	17	53	75.7%	2	68	97.1%
Average		291	59	83.1%	132	218	62.3%	59	291	83.1%

T is newsworthy messages, F is not newsworthy messages, PT is precision of newsworthy and PF precision of not newsworthy.

Table A.5: Confusion matrix of the experiment on the labeling tool

	Newsworthy	**Not Newsworthy**	**Precision**
Newsworthy	291	59	83.1%
Not Newsworthy	182	518	74.0%
Recall	61.5%	89.8%	

Accuracy	77.0%
F-measure	70.7%