Natural Language Processing Technologies for Document Profiling

Antonio Guillén, Yoan Gutiérrez and Rafael Muñoz
Department of Software and Computing Systems
University of Alicante, San Vicente del Raspeig (Alicante), Spain
{aguillen, ygutierrez, rafael}@dlsi.ua.es

Abstract

Nowadays, search for documents on the Internet is becoming increasingly difficult. The reason is the amount of content published by users (articles, comments, blogs, reviews). How to facilitate that the users can find their required documents? What would be necessary to provide useful document meta-data for supporting search engines? In this article, we present a study of some Natural Language Processing (NLP) technologies that can be useful for facilitating the proper identification of documents according to the user needs. For this purpose, it is designed a document profile that will be able to represent semantic meta-data extracted from documents by using NLP technologies. The research is basically focused on the study of different NLP technologies in order to support the creation our novel document profile proposal from semantic perspectives.

1 Introduction

The Internet provides large amounts of information through many types of documents. Users require an easy way to filter these documents to find out the most appropriate documents for their interests, capabilities and needs. These documents also can be needed by other entities for market studies, classify and index documents, detect fake information or illegal activities on the Web. These aspects can be treated by the study of NLP areas, tasks, methods and tools.

In this paper, we present a study on some NLP tasks to determinate which NLP technologies are interesting to extract and obtain relevant information from a document. In this study, we want to address which technologies are available currently, estimate its automation and reliability degree, the problems that can be found on applying them and the most appropriate document's extension. Also, we describe our novel document profile proposal.

A document can be defined as short or long unity of information, principally obtained from websites (a user post, a press article, a comment, a review, etc.). The aim of this study is to investigate the different features that can be extracted by means NLP technologies able to provide enough information for setting up useful meta-data.

For addressing this study we organized the article into two relevant parts. The first part, Section 2, explains what we want to accomplish describing our novel document profile proposal. The second part, Section 3, addresses a study of selected NLP tasks and technologies which serve for supporting the proposal. Finally, Section 4 exposes the conclusions and future works.

2 Document Profile Proposal

Our proposal mainly consists of designing a document profile able to represent different features extracted once NLP technologies are applied on documents, assisted or not by humans. Figure 1 provides a brief overview. As can be seen, there are many features that can only be extracted automatically from documents by using NLP technologies. This work pretends to unify most of these NLP technologies as a whole able to characterize documents from different points of view.

2.1 Documents and NLP Areas

As a first approach, this work will be focused on English documents from the Internet. This is due to NLP technologies have been mostly developed to cover this language and which makes easier to find out NLP tools. Nevertheless, the overall strategy is still valid for other languages. The principal documents addressed are those mostly available on the Internet[1]. Those are press media, social posts, product/service reviews, blogs or per-

[1] http://www.quantcast.com/top-sites

Proceedings of Recent Advances in Natural Language Processing, pages 284–290,
Varna, Bulgaria, Sep 4–6 2017.
https://doi.org/10.26615/978-954-452-049-6_039

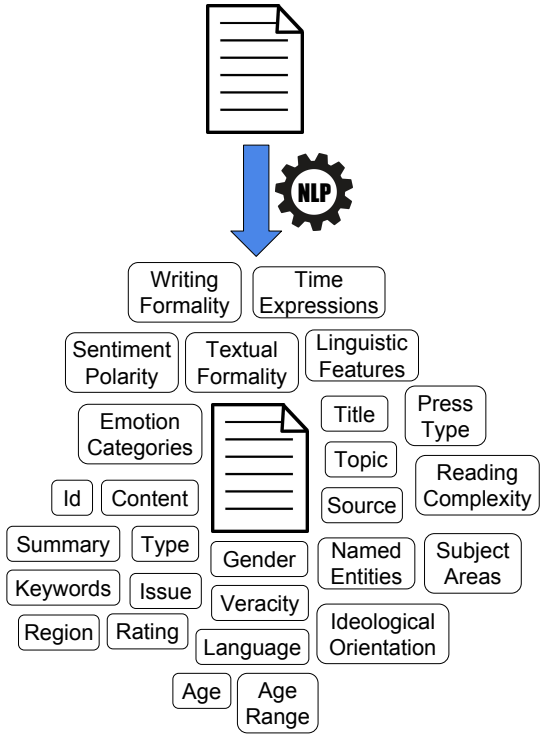

Figure 1: Document profile concept.

sonal websites, academic/science papers, tutorials or instruction manuals, etc. The concrete selection of documents is shown in this schema[2]. Is defined two types of documents (leaf and conceptual), leaf document represents concrete documents that can be obtained the profile, and the conceptual document is defined as group documents that share common features.

For some NLP areas, different NLP tasks require a long or short size document for better results. In our research, a short document is defined as text with 500 words or less, and a long document as texts with 501 words or more. In this research is performed a study of correspondence between short and long documents depending on the NLP task or area. For example, Author Identification tasks give better effectively on long documents, but also some models can give good effectively in short texts (Shrestha et al., 2017) (Green et al., 2013). In the case of Sentiment Analysis, this requires preferably short documents to better focus the different tasks involved, for a long document is necessary to focus on sentence-level or use specific techniques (Basiri et al., 2014). Data Mining requires long text to process in order to reveal common patterns that in single texts are irrelevant. In general, is highly recommended to process long documents in very similar tasks, like text clustering (Ingaramo et al., 2008). An alternative for getting a brief vision of long documents is generating a summary. At this way, long documents would be represented a minor set of phrases. This idea will be considered by us in such cases where the NLP technology to use requires processing short documents instead of long documents.

2.2 NLP Tasks Selection

In our research, a selection of NLP tasks[3] for designing the profiling meta-data properties is carried out. This list has been taken into consideration to be included in our novel document profile proposal. At this way, it can be obtained a long quantity of meta-data contributing to fill document profiles. As can be seen, we have chosen some of the most active NLP tasks nowadays. So, the main goal at this stage is find out friendly access NLP technologies (i.e. tools, APIs, demos, etc) to test and simulate our proposal in a real scenario. At this way, it would be demonstrated the viability of our proposal, not just providing a document profiling scheme.

2.3 Meta-data Properties

The meta-data properties defined for our proposal can be found in the following link[4]. Notice, that each of these properties indicates the NLP technology acronym from which it is obtained. Is included a brief description of each meta-data property.

2.4 Document Profiling Algorithm

This section presents briefly an algorithm to generate document profiles using the available NLP technologies. Algorithm 1 shows the steps to generate a profile from a Web document. The steps are specified as follows. Firstly, the Web document content is extracted. Then a short version of the document using the summarising technology is generated. It is followed by the detection of the document type. It is generated an initial profile from scratch, only including the meta-data obtained currently (i.e. content, summary and type). Next specific meta-data properties for this document type are obtained as is mentioned in Figure 1. Finally, the results compose the document profile.

[2]http://ow.ly/pTaR30dWTj7

[3]http://ow.ly/MPO130e24Jr

[4]http://ow.ly/tQos30dNVcS

Algorithm 1 Document Profiling

Require: *url*, Web document url

```
cont ← getContent(url)
sum ← getSummary(cont)
type ← getType(cont)
profile ← newProfile(cont, sum, type)
listProperties ← getProperties(type)
for each prop ∈ listProperties do
    nlpTech ← prop.getNlpTech()
    value ← nlpTech(cont, summ)
    profile.add(prop, value)
```

2.5 A Document Profile Example

In order to figure out how would be a document profile, we have prepared an example considering a document taken from the CNN website. The document is a real news article about *Everest's climber George Mallory*. He's the first person who tried climb to summit the Everest. He was disappeared and his body was found 75 years later. Expert people tried to discover if he reached the top of Mount Everest or not. The complete news article it's available in CNN website[5]. The procedure follows as to describe the algorithm 1, based on the technologies presented in Table 1, obtaining the next document profile[6].

A brief description of the features set is commented following: (i) The Id property is auto-generated. (ii) The document type detected is *Press Document*. (iii) *Content*, *Title* and *Issue* meta-data properties can be extracted directly from the article, and the source corresponds to the web URL of article. (iv) The content summary is generated. (v) The topics list represents the most frequently used terms in the text. (vi) The region is obtained analyzing the text, in this case, it talks about the north ridge of Mount Everest located in Tibet (China). About the subject areas detected, we can see *History* because it is a historical news article and *Sports* it talks about climbing. (vii) The language detected is English. (viii) The article does not presents any rating. (ix) Keywords are the more representative words of the whole text. (x) In this article, it is not detected any ideological orientation because it only talks about a historical fact related to sports. (xi) Sentiment polarity determines that it is a positive news for people interested in history, climbing and the Mount Everest. (xii) It is detected the emotion category *Surprise* because the man's body has been found 75 years later, something unexpected. (xiii) Some time expressions are detected in the text, these expressions have been converted into date format. (xiv) The name entities detected refer persons and locations retrieved in the text. (xv) In this article is not detected any linguistic feature. (xvi) Reading complexity is *Easy* and with *Formal* text and writing. The news article is from a serious press media. The article is feasible to be read by 12 years old people over, since it is a neutral and simplified information. (xvii) The age of author is not predicted due to the high neutrality of the text. However, the gender is predicted as *Female*. (xviii) The press type is classified as *News Article* and its veracity is *Truthful*, taking into account the formality of press media.

[5] http://ow.ly/7JtB305V8z6

[6] http://ow.ly/Pnmj30dNVtQ

3 NLP Technologies Study

This section describes in more detail each NLP task considered in our work and exposes some available technologies: tools, APIs or demos related. In some cases, only methods and algorithms had been found. Mainly works with evaluations and results are presented. In this study, our interests are mainly focused on available technology (Web service/API, programming library) with certain automation degree to be able to be incorporated in future frameworks or prototypes. In the final section, it is exposed a comparison table of some NLP technologies studied in terms of automation and reliability degrees.

Text Classification

Text classification task (TC) consists of identifying the type of document by analyzing its content. The relevance of this task in document profiling is to determinate which type of document is analyzed, depending its type different features would be extracted. For example, Dandelion API[7] categorises plain text on eight categories: business, economy, sports, etc.

Information Extraction

Information Extraction task (IE) as has been addressed by (Vila et al., 2013) is a way to search and obtain text on large volumes of unstructured information to filter relevant information, using regular expressions, rules and patterns. This is use-

[7] http://dandelion.eu

ful for document profiling because many complex meta-data can be obtained directly from the document. Since our work is mostly focused on Web documents there are too many tools like DEiXTo[8] that can extract information from the W3C DOM documents. The problem with these tools is the low-automation degree due to they should be reconfigured.

Topic Recognition

Topic Recognition task (TR) consists of identifying topics in the text. This task is interesting to classify a document in multiple categories or topics and know the different aspects that dealt the document. TextRazor[9] is a Web tool that lists topics from a text. Also, another Web demo is Meaning Cloud[10] that offers many NLP services among which topic recognition is included.

Keyword Extraction

Keyword Extraction task (KE) is the automatic extraction of relevant terms from a document. Unlike TR, KE doesn't intend to know the different aspects that dealt the document, KE extracts terms that best describe the subject of the document. Statistical Keyword Extraction Tool (SKET) (Rossi et al., 2013) is a programming library for extracting keywords from the text.

Named Entity Recognition

Named Entity Recognition task (NER) tries to locate and classify named entities according to different categories like names of persons, organizations, etc. Stanford NER (Finkel et al., 2005) is a NLP technology available as a programming library and evaluated in some scenarios, domains and corpora, that offers useful NER services.

Time Expression Recognition

Time Expression Recognition task (TER) consists of obtaining temporal expressions from texts. This information is useful to extract historical facts in texts or documents. Stanford SUTime (Chang and Manning, 2012) is a tested technology and available to be used as a programming library. An alternative is TIPSem (Llorens et al., 2010) which has been tested and available as API.

[8] http://dexi.io
[9] http://www.textrazor.com
[10] http://www.meaningcloud.com

Automatic Summarization

Automatic Summarization task (AS) obtains a reduced text from a larger text content. It's interesting to obtain a short version of the same document. (Alcón and Lloret, 2015-07) presents a summarization system for various purposes and domains. This system has been evaluated and is available as API.

Domain Detection

Domain Detection task (DD) is part of Semantic Parsing. This task detects the meaning of sentence using probabilistic semantic models. Our work is focused on ISR-WN (Gutiérrez et al., 2016) which is able to detect domains or categories from different resources obtaining and using relevant semantic trees from a text.

Language Identification

Language Identification task (LI) consists of identifying the language. AlchemyLanguage[11] is a Web demo that offers many NLP services for that purpose.

Polarity Classification

Polarity Classification task (PC) is part of Sentiment Analysis. According to (Mohammad, 2016) this consists of determining whether a text in positive, negative or neutral. Many PC approaches are using machine and deep learning which obtains good results. Among others, we mention two relevant works: (Kiritchenko et al., 2014) which presents an approach using supervised statistical machine learning, and Stanford Sentiment (Socher et al., 2013) that is available as a programming library.

Emotion Detection

Emotion Detection task (ED) is part of Sentiment Analysis. This consists of identifying some emotions expressed in texts. The following set of the basic emotional categories proposed by (Ekman, 2005) are included in this work: Anger, Disgust, Fear, Joy, Sadness and Surprise. One issue of this task is the lack of annotated corpus for evaluation. ToneAnalyzer[12] is a web demo of multiple NLP APIs, including the emotion detection task with 5 Ekman categories: Anger, Disgust, Fear, Joy and Sadness.

[11] http://alchemy-language-demo.mybluemix.net
[12] http://tone-analyzer-demo.mybluemix.net

Readability Analysis

Readability Analysis task (RA) according to (Martín Valdivia et al., 2014), the detection of RC consists of determining documents suitable for being read by specific people age ranges. This includes determining reading difficulties or text comprehension. For addressing this task it is necessary to classify documents on the basis of different levels of reading comprehension. Different measures of readability must be selected, we base our work on the study Flesh-Kincaid Grade Level. This study tries to predict the recommended age to understand the text. The tool Readable.io[13] is a web demo for evaluating the readability level of a text using various grade levels, including Flesh-Kincaid Grade Level.

Informality Analysis

Informality Analysis task (IA) tries to detect the degree informality in texts. This task arises due to the necessity of processing in a personalized way non-traditional textual sources existing on the Internet (i.e. blogs, forums, etc.). TENOR (Mosquera and Moreda, 2012) provides functionalities aligned to this task.

Age Estimation

Age Estimation task (AE) is part of the Author Profiling area. According to (Rangel and Rosso, 2016) AE tries to predict some aspects of authors like age or gender. This task will contribute to our research the discovery of various authors types depending on age ranges. Age Analyzer[14] is a web API that provides functionalities aligned to this task.

Gender Detection

Gender Detection task (GD) is strongly related to AE task, but in this case, it tries to detect the gender of text's author. Gender Analyzer[15] is appropriated to this task as has been before mentioned.

Irony Detection

Irony Detection task (ID) tries to detect if a literal message has an opposite meaning, without a negation marker. The difficulty resides the absence of face-to-face contact and vocal intonation. In the automatic detection of irony is used sentiment analysis, information extraction or decision making to obtain textual features for recognizing irony. (Reyes et al., 2013) presents a research for irony detection in Twitter short documents, using the tasks mentioned above.

Ideology Detection

Ideology Detection task (IDD) tries to detect the ideology orientation expressed in a text content based on a set of opinions or beliefs. Usually, this refers to a set of political beliefs or a set of ideas that characterise a particular culture. (Iyyer et al., 2014) presents a research work where political ideology orientation is detected using neural network technologies.

3.1 Automation and Reliability Study

In the present study, the high-reliability degree is defined as technologies that have an evaluation with high scores of performance. Similarly, the high-automation degree is defined as the type of technology easier to implement or use in each case. For example, Web Services, Java or Python libraries, Algorithms, Web application/demo and Desktop tools. Web Services or online APIs present a very-high-automation degree because it is easy to use them in frameworks or meta-tools developments. Java or Python libraries have a high-automation degree, because it is easy to include them in frameworks or meta-tools develops, but sometimes it is needed to proceed with an adaptation stage of the target programming framework. Algorithms present a medium-automation degree because it should be considered the efforts of reproducing them. Web application/demo tools have a low-automation degree because initially it is difficult to automatically include them, however, an alternative is using web crawling. Most experimental approaches that make use of Web application/demo tools apply semi-automatic procedures. Desktop tools have a very-low-automation degree because it is very difficult to include them in frameworks or meta-tools develops, most of the time the procedures are performed by hand. Table 1 shows a detailed comparison at respect. The automation of document profiling procedures, such as this study reveals, is supported by the reliability degree presented in Table 1. The performing scores have been taken from the bibliography before cited, being them the best technologies found in the state of the art. The technologies where is set "Not found", probably it is because they represent developments related to companies or pri-

[13] http://readable.io
[14] http://ageanalyzer.com
[15] http://www.genderanalyzer.com

NLP task	Technology	Measure	Score	Type
Text Classification	Dandelion	Not found	Not found	Web demo
Information Extraction	DEiXTo	Not found	Not found	Desktop
Topic Recognition	TextRazor	Not found	Not found	Web demo
	MeaningCloud	Not found	Not found	Web demo
Keyword Extraction	SKET	F1	0.7	Java library
Time Expression	Stanford SUTime	F1	0.92	Java library
	TIPSem	F1	0.85	Web service
Named Entity	Stanford NER	F1	0.8876	Java library
Summarising	Summarise	F1	0.57797	Web service
Domain Detection	ISR-WN	F1	0.52	Web service
Language Identification	Alchemy Language	Not found	Not found	Web demo
Polarity Classification	Kiritchenko et al. (2014)	F1	0.855	Algorithm
	Stanford Sentiment	Accuracy	0.854	Java library
Emotion Detection	Zhang et al. (2017)	F1	0.56	Algorithm
	Tone Analyzer	Not found	Not found	Web demo
Readability Analysis	readable.io	Not found	Not found	Web demo
Informality Analysis	TENOR	Not found	Not found	Algorithm
Age Estimation	Age Analyzer	Not found	Not found	Web service
Gender Detection	Gender Analyzer	Accuracy	0.95	Web service
Irony Detection	Reyes et al. (2013)	F1	0.76	Algorithm
Ideology Detection	Iyyer et al. (2014)	Accuracy	0.702	Algorithm

Table 1: NLP technologies comparisons.

vate research. However, it is interesting to study them for future evaluations. At respect to Web demos, these offer limited NLP services which could be possible resolve through business registration or payment of services. Regarding the type "Algorithm", in some cases, this one is not available in the reference papers. Nevertheless, it is considered because the authors could be contacted in some way.

4 Conclusion and Future Work

In this paper, we presented the study of useful NLP technologies to automate the process of building document's profiles. The study revealed that many NLP technologies are interesting to this aim, however, many of them are difficult to be reused. This depends on the licenses of use, visibility, replicability of their algorithms, etc. Another issue is the lack of annotated corpora to evaluate the technologies involved. Clearly, knowing the difficulties found to reuse the NLP technologies will help us to be more focused on considering those technologies with high-automation degree instead of high-reliability degree.

As result, in this paper we demonstrated that many different NLP technologies can converge all in a unique ecosystem, in our case for profiling documents, to be able to provide advanced sights about documents. Based on this result (a document profile), it can be facilitated searching documents and even be able to recommend documents to users taking into account different perspectives never before considered.

As future work, we plan to create a dataset for further supporting the creation and evaluation of document profile. In addition, the study of the results of evaluating different types of document profiles (i.e. social, press, book, etc.) will be included in our research agenda.

Acknowledgements

This work is funded by the University of Alicante (UAFPU2015-5999 and GRE16-01), Generalitat Valenciana (PROMETEOII/2014/001), the Spanish Government (TIN2015-65100-R, TIN2015-65136-C2-2-R) and Ayudas Fundación BBVA a equipos de investigación científica 2016 (ASAP).

References

Óscar Alcón and Elena Lloret. 2015-07. Estudio de la influencia de incorporar conocimiento léxico-semántico a la técnica de análisis de componentes principales para la generación de resúmenes multilingües.

M E Basiri, A R Naghsh-Nilchi, and N Ghasem-Aghaee. 2014. Sentiment prediction based on dempster-shafer theory of evidence. *Mathematical Problems in Engineering* 2014. https://doi.org/10.1155/2014/361201.

Angel X. Chang and Christopher D. Manning. 2012. Sutime: A library for recognizing and normalizing time expressions. In *In LREC*.

Paul Ekman. 2005. *Basic Emotions*, John Wiley Sons, Ltd.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, Stroudsburg, PA, USA, ACL '05, pages 363–370. https://doi.org/10.3115/1219840.1219885.

Rachel M Green, John W Sheppard, Jim Cramer, Lauren Young, Josh Brown, and Ben White. 2013. Comparing Frequency- and Style-Based Features for Twitter Author Identification. *Proceedings of the Twenty-Sixth International Florida Artificial Intelligence Research Society Conference* pages 64–69.

Yoan Gutiérrez, Sonia Vázquez, and Andrés Montoyo. 2016. A semantic framework for textual data enrichment. *Expert Systems with Applications* 57:248–269.

Diego Ingaramo, David Pinto, Paolo Rosso, and Marcelo Errecalde. 2008. *Evaluation of Internal Validity Measures in Short-Text Corpora*, Springer Berlin Heidelberg, Berlin, Heidelberg, pages 555–567. https://doi.org/10.1007/978-3-540-78135-6$_4$8.

Mohit Iyyer, Peter Enns, Jordan Boyd-Graber, and Philip Resnik. 2014. Political ideology detection using recursive neural networks. In *Association for Computational Linguistics*.

Svetlana Kiritchenko, Xiaodan Zhu, and Saif M. Mohammad. 2014. Sentiment analysis of short informal texts. *J. Artif. Int. Res.* 50(1):723–762.

Hector Llorens, Estela Saquete, and Borja Navarro. 2010. Temporal expression identification based on semantic roles. In *Proceedings of the 14th International Conference on Applications of Natural Language to Information Systems*. NLDB'09, pages 230–242.

María Teresa Martín Valdivia, Eugenio Martínez Cámara, Eduard Barbu, Luis Alfonso Ureña López, Paloma Moreda Pozo, Elena Lloret, et al. 2014. Proyecto first (flexible interactive reading support tool): Desarrollo de una herramienta para ayudar a personas con autismo mediante la simplificación de textos .

Saif M. Mohammad. 2016. Sentiment analysis: Detecting valence, emotions, and other affectual states from text. In Herb Meiselman, editor, *Emotion Measurement*, Elsevier.

Alejandro Mosquera and Paloma Moreda. 2012. Tenor: A lexical normalisation tool for spanish web 2.0 texts. In *Text, Speech and Dialogue - 15th International Conference (TSD 2012)*. Springer.

Francisco Rangel and Paolo Rosso. 2016. On the impact of emotions on author profiling. *Information Processing and Management* 52(1):73–92. https://doi.org/10.1016/j.ipm.2015.06.003.

Antonio Reyes, Paolo Rosso, and Tony Veale. 2013. A multidimensional approach for detecting irony in twitter. *Language Resources and Evaluation* 47(1):239–268.

R.G. Rossi, R. M. Marcacini, and S. O. Rezende. 2013. Analysis of statistical keyword extraction methods for incremental clustering.

Prasha Shrestha, Sebastian Sierra, Fabio A González, Paolo Rosso, Manuel Montes-y Gómez, and Thamar Solorio. 2017. Convolutional neural networks for authorship attribution of short texts. *EACL 2017* page 669.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Stroudsburg, PA, pages 1631–1642.

Katia Vila, Antonio Fernández, José M. Gómez, Antonio Ferrández, and Josval Díaz. 2013. Noise-tolerance feasibility for restricted-domain Information Retrieval systems. *Data and Knowledge Engineering* 86:276–294. https://doi.org/10.1016/j.datak.2013.02.002.