LABAS-TS
A System for Assisting Labeling of Training Sets for Text Classification

Alejandro Sierra-Múnera[1,2], Alexandra Pomares-Quimbaya[1], Rafael Andrés González Rivera[1], Julián Camilo Daza Rodríguez[2], Oscar Mauricio Muñoz Velandia[1,2] and Angel Alberto Garcia Peña[1,2]
[1]*Pontificia Universidad Javeriana, Cra. 7 #40-62, Bogotá, Colombia*
[2]*Hospital Universitario San Ignacio, Cra. 7 #40-62, Bogotá, Colombia*

Keywords: Text Classification, Training Set, Labeling.

Abstract: Most text classification techniques rely on the existence of training data sets that are required to build models. However, in many text classification projects, the availability of previously labeled texts is not frequent due to differences in language (e.g. Spanish), domain (e.g. healthcare) and regional or institutional written culture (e.g. specific hospital). In order to contribute to dealing with this problem, this paper presents LABAS-TS, a web-enabled system for assisting the open, collaborative labeling of training sets for text classification. LABAS-TS is framed within a named entity recognition approach that identifies important entities from a domain-specific corpus, based on gazetteers, and uses a language specific sentence analyzer that extracts the portions of text that should be annotated. LABAS-TS was evaluated in the generation of training data sets to classify whether an electronic health record text contains a diagnosis, a test or a procedure, and demonstrated its utility in reducing the required time for building a reliable training set, with an average of eleven seconds between two labels.

1 INTRODUCTION

In this paper, we propose a web-enabled system for assisting the open, collaborative labeling of training sets for text classification. This stems from an increasing interest and use of text mining and natural language processing tools and techniques which aid in finding patterns, clusters or classifications within a diverse corpus of texts (Aggarwal & Zhai, 2012; Naik et. al., 2015). Specifically, machine learning is being increasingly used to generate models that can automatically process texts to recognize entities of interest. Machine learning most often requires an initial data set for training the model with known cases. Each case consists of an input object and an output variable (also called class or objective variable). It follows that in order to build high quality models, training sets must have sufficient cases for each one of the possible values of the output variable.

Specifically, for text mining classification, input objects are documents and the output variable contains the observed values, such as sentiment, opinion or category. Although the ideal case is to have the training data set available in advance, that is not the usual scenario in text mining due to language and/or domain dependence of texts. For instance, if you want to create a model based on medical notes, it must be taken into consideration that they may have different patterns according to the specialty, experience, institution and country of the writers. Even if texts share the same language, an effective global training set for a categorization problem (e.g. emotion analysis) is not viable.

In particular, the LABAS-TS system stems from the need to advance the use of text mining in supporting automatic classification of electronic health records (EHRs). Such records contain a variety of both structured and unstructured fields that can later be analyzed in order to aid with diagnosis, to detect population patterns, to predict possible outcomes, to calculate adherence to medical guidelines, or to extract valuable hospital administration analytics, among others.

However, medical texts are often created under time-pressure, with a variety of both highly codified as well as informal, contextualized terms. In addition, the meaning of the texts can be riddled with ambiguity when a medical professional records suspicions, potential diagnosis, discarded conditions, family history, among others (Dehghan, 2013).

Sierra MÃžnera A., Pomares Quimbaya A., AndrÃĺs GonzÃąlez Rivera R., Daza RodrÃ■guez J., MuÃśoz Velandia O. and Garcia PeÃśa A.
LABAS-TS - A System for Assisting Labeling of Training Sets for Text Classification.
DOI: 10.5220/0006504901740180
In *Proceedings of the 9th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (KMIS 2017)*, pages 174-180
ISBN: 978-989-758-273-8

Moreover, when EHRs are in Spanish, the existence of previously validated training sets for machine learning is much less developed or extensive (Pomares et. al., 2016).

In order to deal with the aforementioned challenges associated with the necessity of building formal training sets that contain language (e.g. Spanish), domain (e.g. healthcare) and organization (e.g. specific hospital) specific patterns this paper proposes LABAS-TS, a web-enabled system for assisting the open, collaborative labeling of training sets for text classification. To do so, we present related works in Section II. We then go on to present the proposed system in Section III. Section IV presents the implementation and use of LABAS-TS in the healthcare domain. Subsequently, Section V discusses the main results of the system, and finally, Section VI presents some conclusions and suggests avenues for future work.

2 RELATED WORKS

With the large volume of data that is stored in healthcare records, the challenge of utilizing these large datasets efficiently depends upon the adequate processing of both structured and unstructured text. Particularly, the large amount of unstructured text that resides in an EHR, like clinical notes, infirmary notes, or individual medical attention reports, cannot be used directly by machine learning models. However, those unstructured texts can be used with machine learning models as labeled training data, product of manual annotation by medical experts.

Crowd *control* (Cocos, 2017) proposes a crowdsourcing alternative to the expert annotation task for radiology reports. The crowdsourced annotations can provide the same performance in classification as the manual experts. A manual experts label process is a time-consuming task and expensive, and with the large amount of EHR data that is generated much faster than the expert can label an impractical task. With the fact that an individual crowdsourced worker may be less reliable than an expert, the crowdsourced training data contains more correctly labeled data in a mayor size of labeled text than expert training data.

Crowdsourcing for medical image classification (de Herrera, 2014) uses the same approach of crowdsourcing techniques for label training data for medical image classification. The crowdsourced team consist of eight experts from a medical image domain using a label platform (Crowdflower, 2017). Using a small team of crowdsourced workers increases the training dataset size and improves the quality of automatic classification, crowdsourced platform increases the number of people participating for a low cost.

Event Based Emotion Classification for News Articles (Li, 2016) built an emotion linked corpus product of crowdsourcing process that is used in document emotion classification as a whole text using Conditional Random Fields CRF classification. One of the problems in emotions analysis is the lack of training data. Annotated emotion corpus is relatively rare in comparison with other NLP tasks. To surpass the lack of annotated data they used crowdsourcing to obtain annotated reliable data.

Brat (Stenetorp, 2012) introduces an annotation tool for the manually created gold standard annotations for textual data. As annotations are the most time consuming in NLP tasks, they proposed an intuitive annotations web interface that can reduce the annotations time. It could be used in different annotation projects with types of corpus data i.e. Medical, biomedical.

The works mentioned before evidence the fact that medical structured and unstructured data is continuously generated over time, making expert manual labeling tasks a never-ending process for machine learning. Assisted and crowdsourced approaches demonstrate that those tasks can be achieved in less time and for larger volumes of information. Despite the nature of structured and unstructured data in many different fields, those crowdsourcing strategies can expand the training dataset dramatically in less time in comparison to a manual expert labeling task. However, training datasets particularly of medical texts contains ambiguity that can be interpreted differently among different types of medical specialists. There is an opportunity to create a formal training dataset that groups the domain, organizational specific patterns and language using the assisted crowdsourced task approach.

3 LABAS-TS OVERVIEW

As Section II illustrates, for the expansion of machine learning projects, it is necessary to have available training sets that can feed analytical models. Improving and assisting the construction of these training sets is the main objective of LABAS-TS system, which begins with two inputs and then follows a process of finding sentences to classify, by presenting them to experts in charge of the annotation process. Figure 1 presents the general process of the

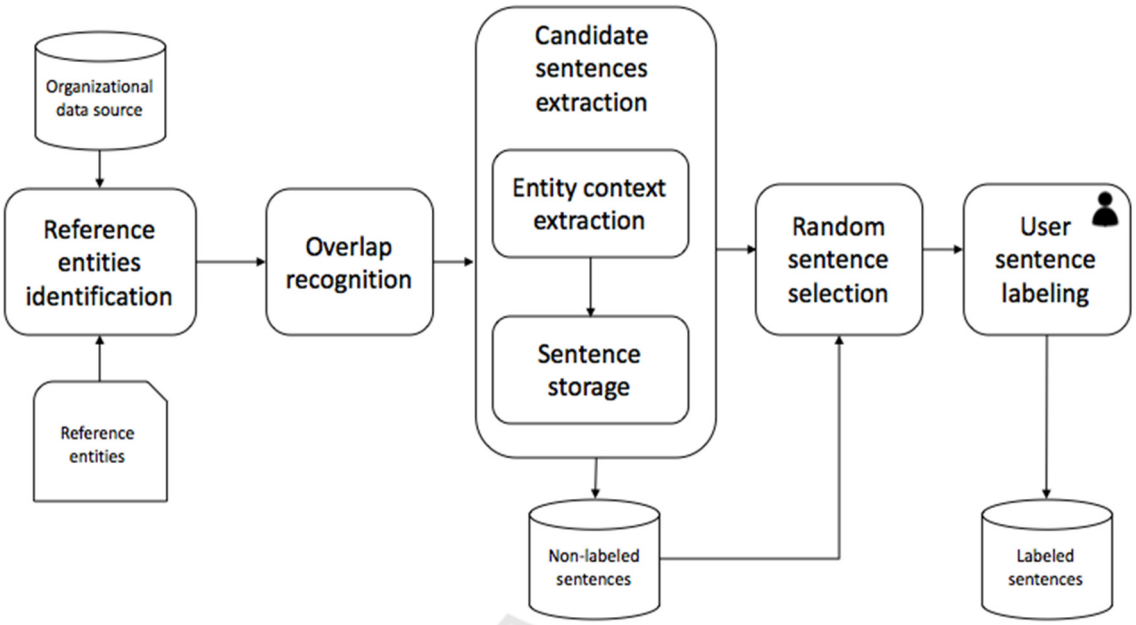

Figure 1: LABAS-TS general process.

system. The first task receives two inputs: the organizational data sources that contain the narrative texts and the lists of reference entities. These entities include the concepts of interest for the machine learning model. For instance, if a model is aimed at identifying opinions or sentiments about a firm, the reference entities should include names of firms or their related brands. The objective of the first task is to recognize these entities within the original organizational data sources. This recognition is not a straightforward activity considering that the words may contain typos or may be expressed using different endings. To deal with this problem, we apply a combined approach for named entity recognition that takes into account these types of cases (Pomares et. al., 2016). The output of the first task is the text annotated with the type of reference entities recognized. Once the system generates the annotations it checks for overlaps between them. If it finds overlap, it selects the most specific annotation, which corresponds to the longest, containing more information. For instance, two entities could be "diabetes" and "type 2 diabetes" and the second one overlaps with the first one, but finding the longest suggest a more specific concept, thus we consider that one.

The next task reads each one of the annotation detected and proceeds to extract candidate sentences surrounding them. For example, the first task found these two entities in a portion of a text. "*...John Doe is a 67 year-old diabetic white male with a history of [COPD], and [hypertension]. Mr. Doe was hospitalized 20 days ago at San Ignacio Hospital for [pneumonia] resulting from [influezna] ...*". In this text four entities were found. From those entities, the following four sentences are extracted:

- John Doe is a 67 year-old diabetic white male with a history of *[COPD]*, and hypertension.
- John Doe is a 67 year-old diabetic white male with a history of COPD, and *[hypertension]*.
- Mr. Doe was hospitalized 20 days ago at San Ignacio Hospital for *[pneumonia]* resulting from influezna.
- Mr. Doe was hospitalized 20 days ago at San Ignacio Hospital for pneumonia resulting from *[influezna]*

Each sentence represents the context of the highlighted term and therefore could have a different class. All the sentences are stored in a repository used by the next task.

Finally, the system selects sentences randomly from the repository of non-labeled sentences and presents them to the expert users who label them using one of the classes required for the machine learning model.

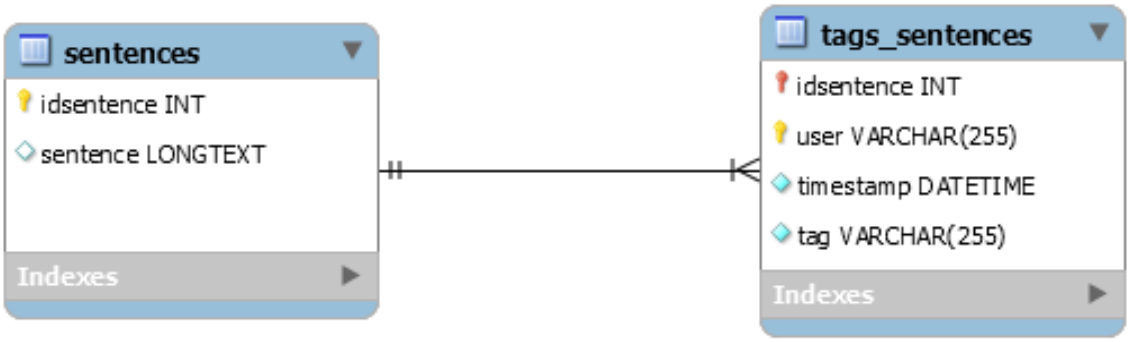

Figure 2: Database Diagram.

In our example those classes could be: confirmed, negated, speculative, pending, patient antecedent or family antecedent. Such classes are highly dependent on the upcoming classification task.

4 IMPLEMENTATION

The first step for labeling a set of sentences corresponds to the process of gathering all the candidate sentences from a corpus. Candidate sentences are those containing reference entities. These entities could be listed from a controlled dictionary or gazetteer, which should contain the different labels related to the list of reference concepts. (Gate.ac.uk, 2017) includes a tool for such activity called "ANNIE Gazetteer". This process has some parameters: if it finds one annotation within another, the behavior should be defined beforehand and defining whether annotations inside tokens are admitted. For instance, the dictionary might contain "Type 1 diabetes" as well as "diabetes". If the first is found, also the second is found, but the parameter *longestMatchOnly* set to true prevents the smaller annotation to prevail. Partial word annotations are discarded. In order to achieve good results in this step it is very important to include openly available gazetteers or ontologies, like the ones available for diagnosis. These sources of information allow direct recognition in the text of the entities whose context is going to be analyzed in the next step. If the organization that is going to use Labas-TS does not have an available gazetteer, it can be built it automatically (Kozareva, 2006) using a corpus of documents that may contain entities, for instance a corpus of non-labeled medical records.

After this step, it finds annotations to all the instances in the corpus of terms within the dictionary we need to extract a context for such annotation. The context is a portion of text containing the annotation and this portion corresponds to all the text a user will be able to read to add the label to the annotation. The context was defined as the sentence containing the annotation. The Spanish Sentence Splitter (Santamaría, 2016) was used to define sentence boundaries. It defines regular expression specific to the Spanish language, which splits the text into sentences. Those should be language specific because a lot of rule exceptions could apply. For instance, a dot is a sentence splitter, but some abbreviation like "hnos." meaning "hermanos" (siblings) should not split a sentence.

Having the annotations and the sentence boundaries we extract all the sentences containing at least one annotation from the gazetteer. If a sentence contains several annotations it is extracted multiple times, one for each annotation contained. This is because one sentence can serve as a different context for several terms, each with a different class.

The sentences are stored in a table (see Figure 2) and each sentence-annotation combination gets an identifier. The remaining process treats them as independent sentences, despite the fact that they might come from the same sentence in the corpus.

Given the new table of sentences, a web application queries the table to get a random sentence to tag. The sentence is shown to the user with a set of buttons, each corresponding to one class (see Figure 3). Right after a user clicks one of the class buttons, the tag is stored in a separate table tags_sentences (see Figure 2).

We mentioned that the sentence was extracted randomly from the sentences table, but it is important pointing out that sentences previously labeled (contained in the tags_sentences table) are ignored, thus only new sentences are shown to the user.

After a significant number of sentences have been labeled we can build a machine learning model. First the tags and sentences tables are retrieved and for each sentence a GATE document is created. The whole text is annotated as one sentence and a feature

Figure 3: Screenshot capturing the page presented to user in which a sentence must be classified.

corresponding to the label assigned by the user is added to the sentence annotation. The implementation of this process was coded as a Java program and all of the GATE documents are stored in xml files.

Next, a GATE Corpus is created from the xml documents and the machine learning algorithm is trained through the Batch Learning PR, containing several classification algorithms such as PAUM, SVM, KNN, Naïve Bayesian Classifier and C4.5. The Batch Learning PR is used in two different moments, training and application. The TRAINING mode, uses the previously created corpus and creates all the files specific for the model. Such model could be exploited in a different moment using the APPLICATION mode, in order to predict the class of an unlabeled sentence.

An important aspect for a training set in machine learning classification problems is having a big number of examples. Therefore, the application is web-enabled in order to reduce the amount of time needed to label and therefore boost the number of labels. The layout of the page presenting the sentence is streamlined, consisting only of the basic instructions of the program, the sentence to be classified and a set of buttons corresponding to the classes (see Figure 3). The sentence is displayed in a bigger font with respect to the other elements and the term of interest is emphasized.

5 RESULTS AND DISCUSSION

As an initial proof of concept of LABAS-TS, the implementation was used as part of a larger project aimed at automatically calculating adherence to medical guidelines in a general hospital. For example, in order to verify whether a procedure has been performed or an exam interpreted, the EHRs for a given set of patients (those corresponding to a specific treatment guideline) must be examined to find significant terms and the potential ambiguity of such terms must be resolved, in our case, through a machine learning text classification model.

Aware of the aforementioned weaknesses in Spanish trained sets, as well as the context-dependent use of a certain amount of terms in the regional and even hospital level, we deployed LABAS-TS for expert medical personnel from our hospital case, by invitation.

The reference terms used in the study case were taken from UMLS Metathesaurus (Bodenreider, 2004). The gazetteer conformed with those terms is the first input of LABAS-TS and controls which entities need to be found to construct the repository of sentences. The other input is the corpus of text from which the sentences are going to be extracted and we used one month of notes from the EHR system used by the hospital. A total of 99,865 sentences where extracted from the text inside the EHRs of the hospital for a given month referencing the entities in the gazetteer. All the sentences were included in the repository.

From those sentences, a total of 1,709 were classified by medical personnel. The time of the experts was limited and one of the goals was to use it in a productive manner and get as many annotations as possible, which means reducing the time spent for each annotation to the minimum. The classes used in our case were Confirmed, Negated, Speculative, Pending, Patient Antecedent and Familiar Antecedent each having a distinct button to facilitate the labeling process. In the instructions section of the user interface, examples were given to clarify the meaning of each class.

After this initial cycle, we gathered comments from the experts involved in the experiment in order to improve user experience and also we analyzed the labels produced, particularly the timestamp data in order to analyze the time it took for them to perform the task.

The users perceived the tool in general as easy to use, the process was quick, and in general the evaluation of each sentence did not take more than one minute. However, it was evident that some characteristics must be guaranteed to facilitate the use. The sentences must be long enough to let the evaluator understand the context adequately. Normally the sentence must be presented with the whole paragraph where it was taken from. Additionally, the simplicity in classifying the sentences depends upon the section of the electronic health record used to extract them. Normally it was easier for the evaluators to make a decision if the sentence came from the analysis sections, and harder if the information was taken from the sections describing the symptoms and antecedents. Finally, it was easier and quicker to classify affirmative sentences (Confirmed class) or negations (Negated class), and harder to define if a diagnosis was a speculation or a familiar antecedent. In those cases, the context information had to be more complete.

Measuring the time spent in between two tags we could check how fast a user can put labels on the sentences. In our measures, the median of the gap between 2 labels is 7 seconds. The average is 11.1 seconds, taking into account just the gaps under 5 min. Some gaps are bigger, but are not taken into account for the average because they correspond to long time periods when the users stopped labeling without logging out.

Our results show that randomly choosing the sentence to be labeled from the candidate sentences, results in an imbalanced dataset, favoring the most popular classes with more examples than the minority classes. Machine learning classifiers could perform better with a more balanced dataset. Therefore, we present strategies that could balance the resulting dataset.

The field of Information Retrieval provides tools for ranking documents according to queries. Ad-hoc retrieval finds the most relevant documents with respect to a given query. Usually a vector space model is used to measure the similarity between the queries and the documents.

Additionally, relevance feedback (Manning et. al, 2009) considers some documents labeled as relevant by the user and adjust the query to rank higher the documents, similarly to the previously relevant documents. In other words, it learns from the positive feedback to consider more information in a new request to improve the results.

In our initial scenario, we want to rank the unlabeled sentences to favor the most unlabeled classes, with the intention of presenting those candidates to the user and balancing the dataset. We could consider the ranking as an ad-hoc retrieval problem where we give some feedback using the already annotated sentences.

The first strategy would use the already labeled sentences from the less common class as positive feedback and rank the unlabeled sentences with such feedback. The results should contain sentences similar to the feedback, presenting the user sentences more likely from the minority class. This strategy requires that at least one sentence is labeled as the minority class and in some cases, that could take enough random annotations that the dataset is already very imbalanced. Sometimes the minority classes could have a known pattern. For instance, if we a considering the class “speculation” we don’t need training to know that sentences containing the word s*uspicion of* are good candidates. Using those known patterns in the first rounds of labeling could guarantee that minority classes have some labeled sentences.

Another strategy involves using the majority classes labeled sentences as negative feedback. Consequently, the ranked sentences would differ from those and belong likely to the minority classes. Negative feedback is usually harder to implement with information retrieval software because they usually provide functionality for positive feedback in the form of “more like this” queries, but no functionality for negative feedback.

6 CONCLUSIONS AND FUTURE WORK

This paper presents a tool for generating classification

datasets to be used mainly by Machine Learning algorithms, in order to predict classes given a set of examples. The goal was to build a software which could allow a group of experts to label a set of automatically-extracted set of sentences with a given set of classes. We describe the extraction process as well as the labeling process, which was designed to make the task uncomplicated and therefore obtain the largest dataset possible.

Proof-of-concept results show that the process was very fast and easy for the user in most cases. Average label time shows that labeling a big dataset could be possible with a reasonable size team of experts.

Some drawbacks were identified in the exercise, mostly regarding the balancing of the classes, and a discussion of how this could be resolved using information retrieval and relevance feedback, suggesting future work.

In addition, we are going to study length of the context presented to the user. Some of the experts felt they needed more context than just a sentence to make a proper classification and expanding to the paragraph level could ease even more the labeling task.

Finally, the tool could be expanded to other annotation tasks like annotating named entities in sentences or chunk detection to detect the structure of the text. The entities that sponsor this project hope to leave LABAS-TS available in its next version for the community.

ACKNOWLEDGEMENTS

This work is part of the project entitled “Sistema de análisis de indicadores de adherencia a las guías de práctica clínica ” funded by Hospital Universitario San Ignacio and Pontificia Universidad Javeriana.

REFERENCES

Aggarwal, C. C., & Zhai, C. (2012). A survey of text clustering algorithms. *Mining text data (pp. 77-128) doi:10.1007/978-1-4614-3223-4_4.*

Bodenreider, O. (2004). The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Research, 32(90001).*

Cocos, A., Qian, T., Callison-Burch, C. and Masino, A.J., 2017. Crowd control: Effectively utilizing unscreened crowd workers for biomedical data annotation. *Journal of Biomedical Informatics*, *69*, pp.86-92.

CrowdFlower. 2017. AI for your business. (online) Available at: https://www.crowdflower.com.(Accessed 12 Jun 2017).

de Herrera, A.G.S., Foncubierta-Rodríguez, A., Markonis, D., Schaer, R. and Müller, H., 2014. Crowdsourcing for medical image classification. *Swiss Medical Informatics*, *30*.

Li, M., Wang, D., Lu, Q. and Long, Y., 2016. Event Based Emotion Classification for News Articles. *PACLIC 30, p.153.*

Dehghan, A., Keane, J.A. and Nenadic, G., 2013, October. Challenges in clinical named entity recognition for decision support. In Systems, Man, and Cybernetics (SMC), *2013 IEEE International Conference on (pp. 947-951)*. IEEE.

Gate.ac.uk. (2017). GATE.ac.uk. (online) Available at: https://gate.ac.uk/ (Accessed 12 Jun. 2017)

Manning, C.D., Raghavan, P. and Schütze Hinrich (2009). *Introduction to information retrieval*. Cambridge: Cambridge University Press.

Naik, M. P., Prajapati, H. B., & Dabhi, V. K. (2015). A survey on semantic document clustering. *Paper presented at the Proceedings of 2015 IEEE International Conference on Electrical, Computer and Communication Technologies, ICECCT 2015,* doi:10.1109/ICECCT.2015.7226036.

Pomares, A., Sierra, A., González, R.A., Daza, J.C., Muñoz, O.M, García, A.A. and Labbé, C., 2016. Named Entity Recognition Over Electronic Health Records Through a Combined Dictionary-based Approach. *Procedia Computer Science, 100, pp.55-61.*

Santamaría, V. (2016) Spanish NLP Tools for GATE. (online) SourceForge. Available at: https:// sourceforge.net/projects/nlptools-es/ (Accessed 12 Jun. 2017).

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S. and Tsujii, J.I., 2012, April. BRAT: a web-based tool for NLP-assisted text annotation. *In Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics (pp. 102-107).* Association for Computational Linguistics.

Sun, C., Rampalli, N., Yang, F. and Doan, A., 2014. Chimera: Large-scale classification using machine learning, rules, and crowdsourcing. *Proceedings of the VLDB Endowment*, *7*(13), pp.1529-1540.

Kozareva, Z. (2006). Bootstrapping named entity recognition with automatically generated gazetteer lists. *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop on - EACL '06. doi:10.3115/ 1609039.1609041.*