Extracting Relationships and Network Structures from Text

David José Martins Forte
david.forte@ist.utl.pt

Instituto Superior Técnico
Av. Professor Cavaco Silva
2744-016 Porto Salvo, Portugal

ABSTRACT

This paper addresses the challenging information extraction problems of named entity recognition and relation extraction over texts written in the Portuguese and English languages. In this work, I used an existing NER system that uses CRF models and does decoding of new texts based on the Viterbi algorithm, namely the Stanford NER system to create models cabable of recognising entities in texts written in Portuguese. To extract relations present in texts, I propose an unsupervised relation extraction approach, which is essentially an adapted version of an algorithm, known as Snowball, that uses unsupervised learning to extract relations by receiving a small set of seed examples, and automatically generating and evaluating patterns for extracting new tuples with the same relation. In this paper, it is presented a method for extracting a signed social network from meaningful co-occurrences of person names within individual sentences, classifying the co-occurrences according to their semantic polarity orientation. This relations extraction method is also used to extract part of relations between locations from geography and fiction books. I analyze the fundamental challenges involved in the development of a learning-based relation extraction system, and provide extensive experimental results with both Portuguese and English texts, for a wide range of parameter combinations.

Keywords

Sentiment Slot Filling, Named Entity Disambiguation, Relation Extraction, Network Analysis

1. INTRODUCTION

The analysis of quantitative information derived from document collections, such as daily newswire texts, e-book libraries or social media contents, holds an enormous potential to solve long standing problems in a variety of disciplines, through massive data analysis [27]. The nascent research area of literary geography/literary cartography, which aims at visibly rendering complex overlays of real and fictional geographies, can also stand to benefit from computational information extraction approaches. The geography of fiction also follows its own distinct rules, since literature can create any space, without physical restrictions. It belongs to the ambitious goals of literary geography to find out more about those rules and to demonstrate that the spatial dimension of fictional accounts can actually be one key to the understanding of the whole plot behind particular texts. Recently, we have been indeed witnessing an increasing interest on the usage of techniques from the areas of text mining, information extraction and natural language processing (NLP), in applications related to to media analytics in general [21].

Information extraction, in particular, can be divided into several sub-problems, and some of the most relevant are named entity recognition, relationship extraction, and named entity disambiguation. This work addresses all these three tasks, using them together to generate new knowledge about entities that are referenced within texts.

Following on ideas from recent work in the area of named entity recognition, focused on the English language, I addressed issues such as the representation of the text chunks corresponding to entities, the usage of features derived from large volumes of unlabeled text, the usage of external knowledge sources that can be incorporated as features, and how to effectively consider all these aspects within a practical NER system.

This work also involves the extraction of positive and negative evaluative opinions, as expressed by particular persons in a given collection of textual documents (e.g., news articles) about other persons, thus supporting the generation of graphs encoding positive and negative interactions between persons. I used a bootstrapping approach that receives a document collection previously processed by a entity disambiguation phase and two sets of user-provided seeds of entity pairs (i.e., one set representing examples of entities with the relation that we want extract, and one set with the opposite relation), and automatically generates and evaluates patterns for new pairs with the same relation. In this work, I also adapted the previous version of the Snowball system for extracting *part-of* relations from geographic and fiction books. The relations are extracted from meaningful co-occurrences of location names within individual sentences (i.e., co-occurrences that are associated to specific linguistic patterns, commonly used to encode part-of relations).

In order to test the quality of the relation extraction system, I report on a case study involving large-scale extractions from newswire documents related to politics, namely documents published on the *Público* online newspaper, from the year 2009 up to the year of 2013. To evaluate the proposed method for social network extraction, I relied on two different ground-truth datasets that were built automatically with basis on politicians that once sat in the Portuguese national

parliament and that appear in the documents considered, together with their respective party affiliations. To test the system relatively to the relations extracted between locations, I report on a case study involving a geographical book from the United Kingdom [5] to generate patterns and extract relations, and two series of fiction books to extract relations using the previously generated patterns, containing the trilogy of *The Scar*, *Perdido Street Station* and *Iron Council* [15, 16, 17], and the trilogy of *Mervyn Peake, The Illustrated Gormenghast* [19]. To evaluate the proposed method, I evaluate the United Kingdom relations extracted based on a geographical web database called GeoPlanetGeoPlanet[1], that contains all the part-of relations regarding place in the *United Kingdom*.

The rest of this paper is organized as follows: Section 2 presents related work, divided in two task, namely named entity recognition and relation extraction. Section 3 describes the importance of Named Entity Recognition in information extraction. Subsection 3.1 explains the techniques that I used to perform to address the particular task. This section finishes showing the experimental results obtained with the proposed method.Section 4 presents the relation extraction system that I developed to extract support and opposition relations, presenting initially an overview of the system's approach and describing in detail the named entity disambiguation process, and how the system extracts support and opposition relations in order to build signed networks. This section also presents the results obtained when testing this systems with the two ground-truth datasets. Section 5 starts by explaining the differences between the relation extraction system used to extract relations between persons and locations. Subsection 5.2 presents the experimental validation, describing evaluation experiments over English geographic and fiction texts. Finally, Section 5.3 presents the main conclusions from this research, and discusses possible directions for future work.

2. RELATED WORK

This section presents the most important related works concerning the named entity recognition task and the relation extraction task.

2.1 Named Entity Recognition

Named Entity Recognition concerns with the detection of entities (i.e., persons, organizations and locations) mentioned in textual documents. Named Entity Recognition (NER) is an important task in the context of Natural Language Processing (NLP).

Named Entity Recognition (NER) systems have been created by using linguistic grammar-based techniques, as well as with statistical models. Statistical systems typically require a large amount of manually annotated training data.

We can model NER as a task of giving tags to words, much like POS tagging. Most statistical NER systems rely on the so-called IOB encoding, where the classifiers are trained to recognize the beginning (B), the inside (I), and the outside (O) of an entity name. Consider the following example:

> David/**B-PER** Forte/**I-PER** studies/**O** in/**O** IST/**B-ORG** ./**O**

In the example, we have two different types of entities, namely a person and an organization. The first entity is composed by more than one word, and thus the first word is assigned to a B(eginning) tag and the other words are assigned to a I(nside) tag. When the entity is composed by one word, it is always assigned a B(eginning) tag, such as in the second entity. The other words in the example have a tag O(ther), because they are not entities. Another popular tagging scheme is referred to as the SBIEO encoding, where *tokens* are marked with five labels, the three labels of IOB and two more, namely, E(nd) indicating words that appear at the end of a segment corresponding to an entity, and S(ingle) indicating words that form one entity individually. Recently, [20] showed that NER models using the encoding scheme SBIEO usually outperform those that use the simplest coding IOB.

NER systems can be developed through maximum entropy models. The Viterbi algorithm can again be used to find the best sequence of tags for a sequence of words. NER models can use features such as the actual word, the previous word, indicators for if the word is in lowercase or if the first letter is in uppercase, lists of known entities (i.e. if the word is in a list of names, then the probability of this word being a name is very high), etc.

Most previous studies have addressed NER tasks focused on texts written in English, although some developments have been reported in the context of applications to the Portuguese language. State-of-the-art NER approaches uses first-order or second-order statistical models for sequential data, based on the principle of maximum entropy, being known in the literature as *Conditional Random Fields* (CRFs)). In particular, recent work focus on the English language [20], addressing issues such as the representation of textual segments corresponding to entities (i.e., comparing the IOB and the SBIEO encodings for entities as individual word tags), the use of resources based on non annotated text (e.g., features derived from word clustering or derived from information relating to occurrences of capitalized words), the external sources of knowledge that can be incorporated as features in the models (e.g., word lists and dictionaries).

2.2 Relation Extraction

In the past, some previous works have addressed the task of extracting social relations between individuals from text. McCallum et al. explored the use of structured data associated to textual documents, such as email headers, for social network construction [13]. Lee et al. [12] built networks that encode pairwise correlations between members of the 109th United States senate, as measured by co-occurences in Google's search engine results, afterwards measuring node

[1]`https://developer.yahoo.com/geo/geoplanet/`

degree and strength as alternative approaches for estimating the importance of nodes, as well as the maximum relatedness subnetwork, its community structure, and its temporal evolution. Gruzd and Hyrthonthwaite explored the use of postings in discussion forums to study interaction patterns in e-learning communities [6] . Diesner et al. [3] or Shetty and Adibi [22] have explored communication networks build from the well-known corpus of Enron email messages. Merhav et al. [14] extracted networks of relations between entities mentioned in blog posts, starting by the creation of entity pairs, then clustering the entity pairs, and finally labeling the clusters with the nature of the relation. Hassan et al. proposed a method to automatically construct signed social networks from online discussion posts, evaluating the extracted networks through social psychology theories of signed networks [7]. My work is related to this particular line of research, because I employ natural language processing techniques to reveal embedded social structures. Similarly to Hassan et al., I also propose to evaluate my work through well known theories of signed networks, but I instead propose a different approach for performing the actual extraction of the networks, from news documents.

The research reported on this paper, which is somewhat related to the mining of attitude associations between individuals, as they are expressed over newswire documents, also connects to a large body of related work addressing different aspects of the problem of analyzing opinions and sentiments expressed over text [18]. One such line of research concerns with the well-studied problem of identifying the semantic polarity orientation of individual words, i.e. finding indications that the opinion direction associated to the word deviates from the norm [8]. Previous works have also showed that polarized words are good indicators of subjective sentences [26], and that automatically classifying words as either positive or negative can enable the automatic identification of the polarity of larger pieces of text, such as sentences or entire documents [25, 8]. In my work, I use a bootstrapping approach for extracting support or opposition relations between individuals, which uses a small set of user provided seed tuples for both the support and opposition relations, and that automatically generates and evaluates interesting patterns for extracting new tuples.

3. NAMED ENTITY RECOGNITION IN PORTUGUESE TEXTS

This section address the development of an efficient and robust Named Entity Recognition (NER) approach for the Portuguese language, using modern statistical models and relying on machine learning.

3.1 The Proposed Approach

I trained and evaluated different NER models (i.e., first-order or second-order CRF models, using the IOB or SBIEO encodings, and using different sets of features) using the CINTIL[2] corpus of modern Portuguese.

[2]http://cintil.ul.pt/

Sentences	26329
Tokens	537064
Types	42957
Persons	9976
Localizations	5216
Organizations	6366
Miscelaneous	2844

Table 1: Statistical characterization for the NER dataset.

The NER task is usually treated as a classification problem over sequential data, in which the objective is to automatically assign the most probable tag sequence $S =< s_1, s_2, \ldots, s_T >$ to a given sequence of observations $O =< o_1, o_2, \ldots, o_T >$ with length T. Accordingly, each sentence obtained from a given text is treated as a sequence of words (i.e., the observed *tokens*), and the resulting sequence of tags (i.e., the resulting annotations, which classify specific segments of text) encode the entities mentioned in the sentence.

Regarding the statistical models that support sequential classification, we have that an approach based on the principle of maximum entropy, known in the literature as *Conditional Random Fields* (CRFs), is nowadays oftenly used. This approach involves the calculation of the conditional probability for each sequence of output labels, given a sequence of input *tokens* [11]. In CRF models, the conditional probability of a sequence of labels S, given a sequence of it tokens O, is given by the following equation:

$$\mathrm{P}(S|O) = \frac{1}{Z_0}\exp\left(\sum_{t=1}^{T}\sum_{k}\lambda_k \times \mathrm{f_k}(s_{t-2}, s_{t-1}, s_t, O, t)\right)$$

In the above formula, each $\mathrm{f_k}(s_{t-2}, s_{t-1}, s_t, o, t)$ is a function of k features, which are typically binary and return the value zero for all cases except if s_{t-2}, s_{t-1} and s_t take some particular labels, and if the observations also possess certain properties. The weights λ_k associated with each feature function are learned automatically through training with supervision (i.e., based on annotated data). In order to have a conditional probability with a value between 0 and 1, we calculate the normalization factor Z_0 considering all the sequences and all possible labels:

$$\mathrm{Z_0} = \sum_{s}\exp\left(\sum_{t=1}^{T}\sum_{k}\lambda_k \times f_k(s_{t-2}, s_{t-1}, s_t, o, t)\right)$$

The software package used to perform named entity recognition was Stanford NER. The dataset had to be adapted in order to be in the convenient representation format accepted by Stanford NER, where each line has an individual *token* along with the corresponding IOB or SBIEO tag.

The Stanford NER package already provides trained models for the English language, but lacks models for the Por-

tuguese language. Nonetheless, it provides an easy framework to train new models with data from different sources. A NER model based on the formalism of Conditional Random Fields was trained using the part corresponding to the written CINTIL International Corpus of Portuguese, which is composed of 537064 annotated word tokens taken from texts collected from different sources and domains. Table 1 shows a statistical characterization for the corpora used in the evaluation experiments, after preprocessing the CINTIL corpus, presenting the number of *sentences* and *tokens* for each dataset, together with the number of entities of each type.

The considered basic features to train the models were the current words, previous words, and next word, within a window of two tokens, are all used as features in the NER models, and the respective tags of these words. I also used, word shape, prefixes and suffixes of the current word, a check value if the current word appears in the first position of the sentence, and a check value if the first letter of the current word is upper-cased, names lists, brown-cluster, and gazetteers.

In order to use Brown clustering I used an open-source[3] implementation of the Brown [23], together with two large collections of Portuguese texts, namely by using a set of phrases that combines the CINTIL corpus with news published in Público newspaper over a period of over 10 years.

Apart from global features obtained with methods of word clusters, I also held tests involving the introduction of a simple feature that essentially captures the probability of having each word appearing in the texts with the first letter capitalized. Since entities typically correspond to words that often arise with the first letter capitalized in the text, this feature can provide a good indication of whether a type of particular *token* is or not used within entities mentioned.

With regard to external sources of knowledge, I consider gazetteers with names of people, organizations and locations as described in the versions in Portuguese DBpedia. I also used lists of words specific corresponding to the most common first names or family names, collected from various sources on the Web (e.g., from Wikipedia pages showing lists of proper names or family names popular).

After training a classification model, one can then use it to assign the most likely sequence of labels for new *token* sequences (i.e., *decode* the text). Since CRF classifiers produce results in the form of probabilities, which can also be translated into costs, a possibly approach involves the use of dynamic programming (e.g., the Viterbi algorithm) for assigning the label sequence that maximizes the overall probability on the sequence of words, given the consistency of the solution.

3.2 Results

I evaluate the NER experiments, with texts written in Portuguese, through some of the common metrics that have been used in previous works in the area of relation extraction from text. Specifically, I use precision, recall, the F_1 measure, and accuracy.

In order to use Brown clustering I used an open-source[4] implementation of the Brown procedure that follows the description given by [23], together with two large collections of Portuguese texts, namely by using a set of phrases that combines the CINTIL corpus with news published in Público newspaper over a period of over 10 years, inducing thousand clusters of words, as done in previous related work [20, 24].

Apart from global features obtained with methods of word clusters, I also held tests involving the introduction of a simple feature that essentially captures the probability of having each word appearing in the texts with the first letter capitalized. The values associated with this feature can be easily estimated based on a set of unannotated texts (i.e., the same used in the induction of groups of words). Since entities typically correspond to words that often arise with the first letter capitalized in the text, this feature can provide a good indication of whether a type of particular *token* is or not used within entities mentioned.

Table 2 presents the mean value calculated with basis on the results obtained in each one of the 5 folds, using cross-validation with the CINTIL corpus. The results are presented in terms of the various evaluation metrics, and in terms of the quality of results when assigning the IOB or SBIEO tags (i.e., accuracy of individual predictions as well as micro-average precision, recall and F_1 scores), as well as when classifying spans corresponding to named entities (i.e., the precision, recall and F_1 score, as for each entity type, again terms of micro-averaged scores). Since the tests with the CINTIL corpus were made by means of cross-validation, I report the average values calculated with basis on all 5 folds.

4. EXTRACTING SIGNED NETWORKS FROM TEXT

This section describes a relation extraction system, which performs extraction of support and opposition relations between persons in documents written in the Portuguese language. We can derive the networks from meaningful co-occurrences of person names within individual sentences, classifying the co-occurrences according to their semantic opinion polarity orientation.

4.1 The Proposed Approach

The Snowball [1] system uses a bootstrapping approach for extracting relations from natural language text. As shown in Figure 1, Snowball starts with a small set of user-provided seed tuples for the relation of interest (i.e., example tuples for support or opposition relations), and automatically generates and evaluates patterns for extracting new tuples with the same relation. Snowball also uses a strategy for evaluating the quality of the patterns and the tuples that are generated in each iteration of the extraction process, where

[3] `https://github.com/percyliang/brown-cluster`

[4] `https://github.com/percyliang/brown-cluster`

		Evaluation based on Entity Spans																Aval based on Tokens			
		PER			LOC			ORG			MISC			ALL							
Model	Characteristics	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	A	P	R	F_1	A
CRF 1º Order	Basic	81.77	53.70	64.53	75.75	56.09	63.99	61.28	53.31	53.19	70.67	31.70	43.48	77.35	51.89	62.05	77.38	78.04	56.66	65.54	96.05
IOB	+Word Clusters	81.83	53.80	64.64	75.90	56.00	64.00	60.96	53.65	53.40	70.73	31.61	43.39	77.34	52.01	62.12	77.36	78.00	56.70	65.55	96.05
	+Gazettes	83.26	64.39	72.51	70.89	69.88	70.01	62.27	53.96	55.39	70.65	**34.02**	**45.70**	77.41	**60.76**	**68.06**	77.43	78.67	**63.98**	**70.52**	**96.62**
	+Lowercase	83.26	64.39	72.51	70.89	69.88	70.01	62.27	53.96	55.39	70.65	**34.02**	**45.70**	77.41	**60.76**	**68.06**	77.43	78.67	**63.98**	**70.52**	**96.62**
CRF 2º Order	Basic	79.49	52.04	62.52	72.91	54.78	62.00	**74.47**	54.85	**63.05**	68.17	32.98	44.39	75.24	50.48	60.34	75.27	76.50	56.05	64.56	95.48
IOB	+Word Clusters	81.77	53.66	64.50	75.74	56.27	64.11	61.16	53.32	53.27	70.33	31.57	43.27	77.33	51.87	62.02	77.35	78.03	56.61	65.50	96.05
	+Gazettes	83.33	64.31	72.49	71.23	70.02	**70.21**	62.00	52.50	54.86	70.81	33.74	45.45	77.54	60.66	68.05	77.57	**78.79**	63.85	70.50	**96.62**
	+Lowercase	83.33	64.31	72.49	71.23	70.02	**70.21**	62.00	52.50	54.86	70.81	33.74	45.45	77.54	60.66	68.05	77.57	**78.79**	63.85	70.50	**96.62**
CRF 1º Order	Basic	83.39	54.04	65.30	76.59	56.14	64.35	60.30	53.53	52.66	**72.54**	30.80	43.00	77.93	51.92	62.26	77.95	77.18	55.05	64.15	95.98
SBIEO	+Word Clusters	83.37	54.04	65.30	76.63	56.14	64.37	60.43	53.67	52.77	72.15	30.91	43.02	77.93	51.97	62.30	77.95	77.13	55.13	64.19	95.98
	+Gazettes	84.19	**64.55**	**72.99**	70.35	70.27	70.00	61.85	53.92	55.02	72.07	33.27	45.30	77.68	60.53	68.02	77.71	77.19	62.28	68.90	96.54
	+Lowercase	84.19	**64.55**	**72.99**	70.35	70.27	70.00	61.85	53.92	55.02	72.07	33.27	45.30	77.68	60.53	68.02	77.71	77.19	62.28	68.90	96.54
CRF 2º Order	Basic	83.52	53.72	65.13	**76.74**	56.04	64.31	60.55	53.81	53.11	71.60	30.68	42.72	**78.13**	51.86	62.27	**78.14**	77.39	54.86	64.09	95.97
SBIEO	+Word Clusters	83.18	53.87	65.10	76.55	56.33	64.43	60.53	53.37	52.74	71.92	30.85	42.98	77.91	51.83	62.17	77.92	77.14	54.68	63.89	95.96
	+Gazettes	**84.43**	64.48	73.05	70.20	**70.28**	69.95	61.99	**54.96**	55.17	71.53	33.25	45.14	77.70	60.47	67.99	77.72	77.08	62.29	68.86	96.53
	+Lowercase	**84.43**	64.48	73.05	70.20	**70.28**	69.95	61.99	**54.96**	55.17	71.53	33.25	45.14	77.70	60.47	67.99	77.72	77.08	62.29	68.86	96.53

Table 2: Results obtained with the CINTIL corpus, using cross validation with 5 folds.

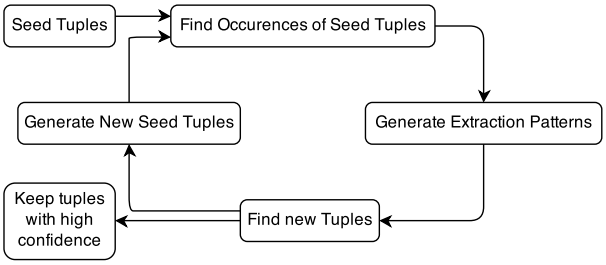

Figure 1: The architecture of Snowball, a partially-supervised information extraction system.

only those that are regarded as being sufficiently reliable will be kept for the following iterations of the algorithm.

In my extension of the Snowball, the system starts with two user-provided set of seed tuples, one for the relation of interest (i.e., example tuples for support or opposition relations), and another for the oposite relation of interest (i.e., example tuples for the oposite relation that we want to extract).

A crucial step in Snowball is the generation of patterns to find new tuples in the documents. In order to generate a pattern, Snowball groups occurrences of known tuples in documents, if the contexts surrounding the tuples are similar enough. More precisely, Snowball generates a tuple for each string where a seed tuple occurs, and then clusters these tuples. Given a set of seed tuples $< e_1, e_2 >$ and having found the text segments where e_1 and e_2 occur close to each other, Snowball analyzes the text that connects e_1 and e_2 to generate patterns. In order to represent each one of these text segments I keep a 5-tuple containing the context before the first entity (left context), the name of entity one (e_1), the context between the two entities (middle context), the name of the second entity (e_2), and the context after the second entity (right context). The algorithm represents the left, middle, and right contexts associated with an extraction pattern as vectors of weighted terms calculated with TF-IDF.

In this work, the patterns are generated using a different clustering approach of the original Snowball, based on DBScan [4].

DBScan requires two parameters, namely (1) a threshold similarity, and (2) the minimum number of instances required to form a cluster. DBScan starts with an arbitrary tuple that has not been visited. This tuple's neighbours are retrieved, and if this set contains a sufficient amount of tuples, a cluster is started. Otherwise, the tuple is labeled as noise. If a cluster has been started, then the neighbours of the previous tuple's neighbours are retrieved and analized as before. The process continues, until the cluster is completely found. Then, a new unvisited tuple is retrieved and processed, leading to the discovery of other clusters.

After generating patterns, Snowball scans the collection to discover new tuples, by matching text segments with the most similar pattern, if any. Each candidate tuple will then have a number of patterns that helped generate it, each with an associated degree of match. Snowball uses this information, together with information about the selectivity of the patterns, to decide what candidate tuples should actually be added to the set of final tuples.

In order to estimate the quality of the patterns, we can weight them based on their selectivity, and trust the tuples that they generate accordingly. Thus, a pattern that is not selective will have a low weight. The tuples generated by such patterns will be discarded, unless they are supported by selective patterns. We also only keep tuples with high confidence. The confidence of the tuple is a function of the selectivity and the number of the patterns that generated it. Intuitively, the confidence of a tuple will be high if it is generated by several highly selective patterns.

As an initial filter, I eliminate all patterns supported by fewer than a specific threshold of seed tuples. I then update the confidence of each pattern, checking each candidate tuple $t =< e_1, e_2 >$ that is generated by the pattern in question. For each candidate tuple, we check if there exists a set of high confidence previously extracted tuples for e_1 (e.g., $< e_1, e_x >$, $< e_1, e_y >$). If e_2 is equal to either e_x or e_y, then the tuple t is considered a correct match for the pattern, and an unknown match otherwise. Additionally, if the candidate tuple matches with a known negative example tuple (i.e., a tuple created by a set of seeds given initially as representing the oposite type of relation), the tuple t is considered a incorrect match for the pattern and the confidence of P is decreased. More formally, Snowball defines Conf(P), i.e, the confidence of a pattern P, as follows:

$$\text{Conf}(P) = \log_2(P_c) \times \frac{P_c}{P_c + P_u \times w_u + P_i \times w_i} \quad (1)$$

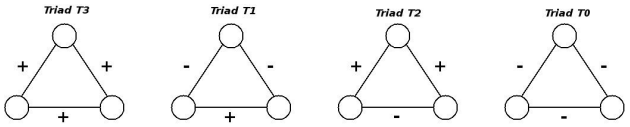

Figure 2: The different undirected signed triads.

In the formula, P_c is the number of correct matches for P , P_u is the number of unknown matches, and P_i is the number of incorrect matches, adjusted respectively by the w_u and w_i weight parameters. The confidence scores are normalized so that they are between zero and one.

The approach used to calculate the pattern's confidence is different from that used in the original Snowball. In the initial approach, the confidence formula simply use the P_c and P_i, where this P_i represent the P_u on the adapted formula.

We can also use another approach to calculate the patterns confidence, using estimated relations. These relations are predicted with basis on social psychology theories, namely the structural balance theory [9], which have been proposed to reason about how different patterns of support and opposition links provide evidence for the expression of different kinds of relationships. This theory has been shown to hold both theoretically and empirically for a variety of social community settings. I argue that by showing that this theory also hold for our automatically constructed network, we can generate new relations. Structural balance considers the possible ways in which triangles can be signed – see Figure 2. The theory posits that triangles with three positive signs (i.e., three mutual friends) and those with one positive sign (two friends with a common enemy) are more plausible, and hence should be more prevalent in real networks, than triangles with two positive signs (two enemies with a common friend) or none (three mutual enemies). Balanced triangles with three positive edges exemplify the principle that *the friend of my friend is my friend*, whereas those with one positive and two negative edges capture the notions that *the friend of my enemy is my enemy*, *the enemy of my friend is my enemy*, and *the enemy of my enemy is my friend*. The structural balance theory has also been developed extensively since the initial proposal, including the formulation of a variant named weak structural balance, proposed by Davis in the 1960s as a way of eliminating the assumption that *the enemy of my enemy is my friend* [2]. In particular, weak structural balance posits that only triangles with exactly two positive edges are implausible in real networks, and that all other kinds of triangles should be permissible.

Based on the idea that triangles with three positive signs and with one positive sign (two friends with a common enemy) are plausible, we form triangle relations composed by three entities where two of relations are know and follow one of the two theories, and then we predict the third relation. For example, if have three entities e_1, e_2, and e_3 where e_1 and e_2 have a support relation, e_2 and e_3 have an opposition relation, then we will say that e_1 and e_3 have a opposition relation. Using these ideas to calculate the pattern's confidence, Snowball defines Conf(P), as follows:

$$\text{Conf}(P) = \log_2(P_c) \times \frac{P_c + E_c \times w_{ec}}{P_c + P_u \times w_u + P_i \times w_i + E_c \times w_{ec} + E_i \times w_{ei}} \quad (2)$$

In this version of the confidence formula, we also use the E_c as the number of correct estimated relationships for P, and E_i as the number of estimated relationships calculated using the opposite seeds given initially (i.e., if one of the three relations in the triangle is originated by the opposite seed set, and one of the theories is verified), adjusted respectively by the w_{ec} and w_{ei} weight parameters.

The confidence of the extracted tuples is calculated as a function of the confidence values for the patterns and the number of patterns that generated the tuples. Intuitively, Conf(t), i.e., the confidence of an extracted tuple t, will be high if t is generated by several highly selective patterns. More formally, the confidence of t is defined as follows:

$$\text{Conf}(t) = 1 - \prod_{i=0}^{|P|} (1 - (\text{Conf}(P_i) \times \text{Match}(C_i, P_i))) \quad (3)$$

In the formula, P is the set of extraction patterns that generated t, and C_i is the context associated with an occurrence of s that matched a specific pattern P_i with degree of match Match(C_i, P_i). The Match(C_i, P_i) value represent the biggest similarity value obtained when we compare the tuple with each one of the pattern's centroid. After determining the confidence of the candidate tuples, the algorithm discards all tuples with low confidence (i.e., those with a score below a given threshold), because these tuples can add noise into the pattern generation process, which would in turn introduce invalid tuples.

4.2 Results

The network extraction method that was outlined in the previous section was applied to newswire corpora in the Portuguese language, namely to news articles published on the online version of *Público*, from the last 5 year. The Portuguese corpus consists of a total of 176.865 Portuguese news stories containing 2.452.713 sentences, maintaining a total of 244.760 different persons (after disambiguation).

Figure 3 shows the regular and cumulative distributions for the total of news articles published in each year, as well as for the number of sentences analyzed by Snowball, the same figure also show the distributions of the total number of person names that are mentioned in those years, after the disambiguation step.

When assessing the quality and veracity of the results for the support and opposition relations extracted, we conducted an empirical analysis and relied on profile information, due to the fact that there are not strict parameters or ground-truth lists to truly assess the relation between persons.In order to evaluate the relations that were extracted, we also built two different ground-truth datasets automatically, based on a list with all the Portuguese politicians which seat in the parliament and that appears in the input documents of *Público*

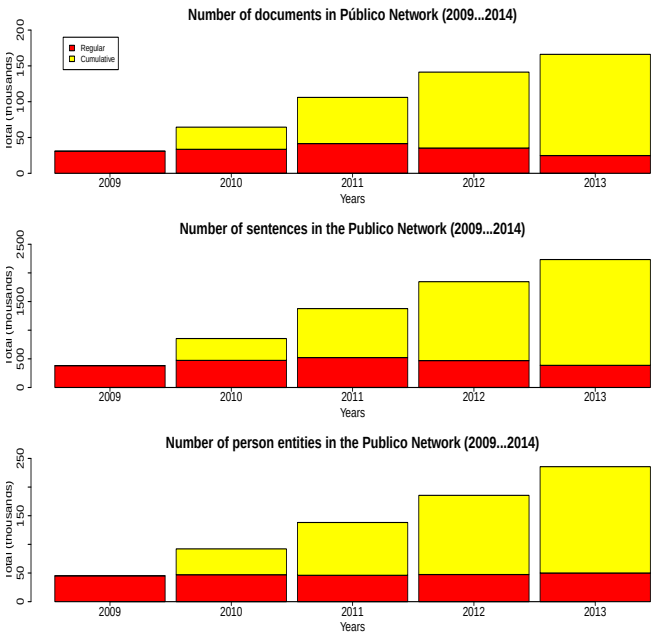

Figure 3: Number of news, sentences and number of person entities in the Publico dataset.

newspaper, and the respective parties to which they belong, assuming that:

1. All the political entities that belong to parties with the same orientation (i.e., left or right political leans) have a support relation between them, and they are assumed to have an opposition relation towards the persons with the other orientation. This dataset is composed by 5264 support relations and 20 opposition relations.

2. All the political entities that belong to the same party have a support relation between them, and an opposition relation towards the persons on the other parties. This dataset contains more relations, with 4401 support relations and 618 opposition relations.

I used the proposed procedure to extract support and opposite relations, performing a extensive set of experiments where I tested the system 350 times. Each one of these tests represent result of combinations of the Snowball parameters, where all have fixed values except four parameters, i.e, min_tuple_confidence, min_degree_match, min_pattern _support, and DBScan_eps. The relations extracted are evaluated through some of the common metrics that have been used in previous works in the area of relation extraction from text. Specifically, I used precision, recall, the F_1 measure, and accuracy.

Figures 4 and 5 present the obtained results for the two ground truth datasets, where each point represents the obtained values in one of the experiments, in terms of the precision and recall measures. The blue point represents the result with higher F_1, with the parameters 0.6 in min_tuple_confidence, 1 in min_pattern_support, 0.6 in min_degree_match, and DBScan_eps with a value of 0.5, for both experiments, returning a F_1 score of 69.26% and 77.83%. respectively. The reason of these results is the small number of opposition relations in the dataset about the left/right ideologies because when evaluating the opposition relations, most of the experience have 1 correct relations extracted (i.e., they do not extracted almost any of the 20 opposition relations) and when the average of the precision is computed, the value of the opposition relations is low and this in fact decrease the overall precision..

	My Snowball	Original Snowball
Number of nodes	588	721
Number of oppositive arcs	376	464
Number of support arcs	1248	1395
Number of arcs	1624	1859
Min Tuple Confidence	0.6	
Min Pattern Support	1	
Min Degree Match	0.6	
DBScan Eps	0.5	

Table 3: Nodes and arcs of the resulting network and the parameters that generated it.

I then used the proposed procedure, with the best performing parameters, to build a signed network from the Portuguese news dataset, afterwards performing a statistical characterization for the the network that was generated. Table 3 presents the number of nodes and relations extracted in the experiment with better results, as well as the parameters used to obtain these results.

Table 4 presents the best evaluation results obtained with both datasets, comparing them with the results obtained with the original version of Snowball, when using the same values for the parameters. Observing the results, one can conclude that the adaptations made on the Snowball system, improved the results when extracting support and opposition relations. It would also be interesting to observe the results if I had a dataset with all the support and opposition relations in the 5 years of *Público* analyzed instead of create two datasets based on theories.

Figure 6 shows a sample of the network corresponding to the entities with more than 20 relations extracted, presenting the disambiguated name of each entity. The blue nodes represent to names of persons and the edge represent the a polarity relation between two persons. The size of each node is proportional to the number of relations of each entity, i.e., if a specific entity have several relations with other entities, then its size will be bigger than a entity that have one or

		Dataset 1	Dataset 2
Original Snowball	Precision	50%	64.49%
	Recall	43.67%	65.87%
	F_1 Measure	46.62%	65.04%
	Accuracy	96.43%	96.17%
My version of Snowball	Precision	65.74%	82.35%
	Recall	73.18%	73.78%
	F_1 Measure	69.26%	77.83%
	Accuracy	96.71%	96.16%

Table 4: Evaluation measures of the two ground truth artificial datasets.

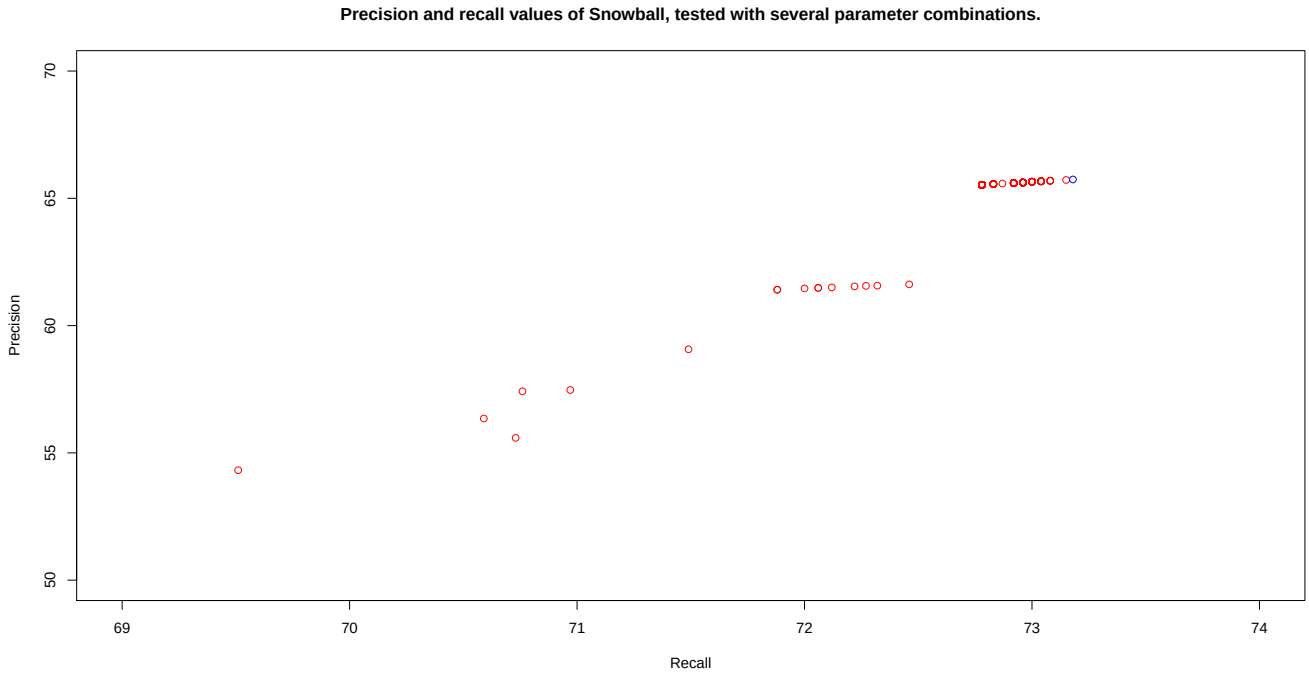

Figure 4: Results obtained by Snowball with various combinations of the parameters when testing with the ground truth dataset based on left/right ideologies.

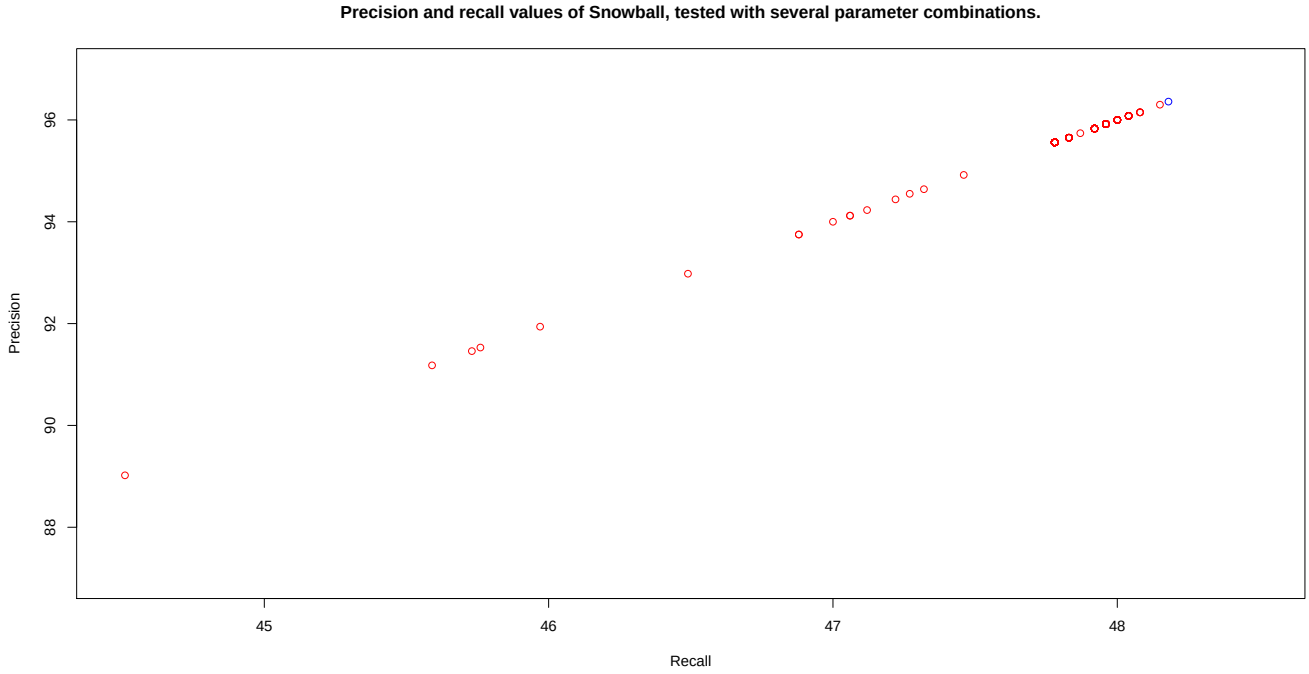

Figure 5: Results obtained by Snowball with various combinations of the parameters when testing with the ground truth dataset based on party affiliations.

few relations. The color of the edge describes the sign of the relation, the green represent a support relation and the red represent a opposition relation.

5. EXTRACTING PART-OF RELATIONS BETWEEN LOCATION REFERENCES

Figure 6: Signed Network with the support and opposition relations extracted from Público.

This section presents a method for extracting part of relations between locations from books. I used basically the same bootstrapping method explained in the previous section with some differences, due to the fact that in this case we want extract directed relations. I report on extractions from English fiction books.

5.1 The Proposed Approach

The main difference is related to the use of direction in the relations, being all the other differences caused by this one. The seeds given initially have direction, i.e a seed $< e_1, e_2 >$ is completely different from a seed $< e_2, e_1 >$. For instance, having a relation London *part of* United Kingdom is different of having United Kingdom *part of* London. Given this, all the seed and extracted tuples need to have the same direction, if e_1 *part of* e_2 then all the other seeds need a direction given by the first entity towards the second. In order to discover more seeds, we compute the transitive closure of the seeds before each iteration.

$$\text{Conf}(P) = \log_2(P_c) \times \frac{P_c}{P_c + P_u \times w_u + P_i \times w_i} \quad (4)$$

Finally, the most important change relatively to the system presented before is the formula for the pattern's confidence calculation. In this case we use a confidence formula different from the one presented in the previous section(Formula 4). In this Formula, where we check for each candidate tuple $t =< e_1, e_2 >$ if these exists a set of high confidence previously extracted tuples for e_1 (e.g., $< e_1, e_x >$, $< e_1, e_y >$). If e_2 is equal to either e_x or e_y, then the tu-

ple t is considered a correct match for the pattern, and an unknown match otherwise.

Although the correct and unknown matches are the same as in the previous system, the incorrect matches are different. If the candidate tuple matches with a known tuple s that was previously extracted, where $s =< e_2, e_1 >$, then the tuple t is considered an incorrect match for the pattern. The confidence of a pattern P, as expressed in Formula 4.

5.2 Results

In order to evaluate the system, I used a geographical book regarding the United Kingdom [5]. This book consists of a total of 6439 sentences with a total of 485 different locations.

I used a specific tool, the Yahoo! GeoPlanet to validate the geographic places and identify the relations between them. In practical terms, GeoPlanet is a resource for managing all named places on Earth. By using GeoPlanet, we can traverse the global spatial hierarchy. In this case, I use the tool to explore the places hierarchy, for instance, for a location *London*, we want the location's parents, i.e. *England*, *United Kingdom*, and *Europe*.

The relations extracted are evaluated through the same common metrics that were used in the evaluation of the relation extraction system explained in the previous chapter. Specifically, precision, recall, and F_1 measure.

I used the proposed procedure to extract *part-of* relations from the English book dataset, in this case also performing an extensive set of experiments where I tested the system 350 times, with the same parameter combinations explained in the previous section.

Figure 7 shows the obtained results, where each point represents the obtained values in one experiment, in terms of precision and recall measures. The blue points represent the 10 results with the higher F_1 scores, and so the best results obtained, being the best result a combination of all the fixed parameters together with a 0.2 value in the min_tuple_confidence, 1 in min_pattern_support, 0.6 in min_degree_match, and DBScan_eps with a value of 0.4, returning a F_1 of 19.2%.

As for the second set of experiments, the main objective is to show that the patterns that extract the relations between the UK places, can be used to extract part-of relations in another context (e.g., fiction books). More precisely, I used a collection of fiction books, composed by the Bas-lag trilogy by [15, 16, 17], namely *The Scar*, *Perdido Street Station* and *Iron Council*, and the Gormenghast trilogy by [19], namely *Titus Groan*, *Gormenghast* and *Titus Alone*, to apply the patterns generated before and extract the *part-of* relations between fictional places, consisting in a total of 14.870 sentences, and a total of 106 different locations.

Table 5 shows a characterization for the relations extracted by the system, where we can seed that few relationships where in fact extracted. Despite this short value of relations extracted, the *unique location pairs* column in the table also contains a short value. This column present all the pairs present in the books, even pairs with no relations (i.e., *London*, *Manchester*, etc.) that are very common in this kinds of books. With this, we can conclude that there were not many relationships to extract, and the number of fiction relations extracted are not bad at all.

5.3 Conclusions and Future Work

In this paper, I have shown that natural language processing techniques can be used effectively to extract signed social networks from newswire documents.

Relatively to the NER experiments, I revisited the development of NER systems for the Portuguese language, using modern statistical models in conjunction with machine learning. In particular, I compared different models and different representations for the mentioned entities, and quantified the impact of considering different characteristics derived from large volumes of non annotated text or external sources of knowledge. My results showed that first-order CRF models, using the IOB encoding, achieved the best results in terms of the quality of assignments. In fact, the overall results showed that IOB encoding is better than the SBIEO encoding, leaving the hypothesis that something went wrong.

In relation extraction, I report on large-scale extractions from newswire documents related to politics, written in Portuguese. To evaluate the proposed method, I used two ground-truth datasets, based on a list with all the Portuguese politicians and their political orientations. Finally, I showed that natural language processing techniques can be used effectively to extract relations between location entities mentioned in books, reporting on extractions from books related to geography and fiction, written in English. In order to evaluate the proposed method, I used a dataset based on a geographical web database called GeoPlanet[5]. For future work, it would be interesting to conduct a detailed analysis of the resulting networks focusing for instance on the identification of network clusters or influential nodes.

Despite the interesting results, that are also many open challenges for future work. It would be interesting, for instance, to experiment with the usage of the sorted neighbourhood method [10] in the named entity disambiguation step. The use of this method will reduce the comparisons between candidate names that will lead to a performance improvement.

The experiments reported in this dissertation have mostly addressed relation extraction between named entity references, assuming the existence of a named entity recognition system. For future work, I believe that the usage of an anaphora resolution system would be interesting, in order to discover sentences in the collection of documents that are currently not being considered because they did not refer directly to any person name. For example, in the sentence ***He*** *criticized* ***him*** *for the way he talked*, we have a opposition relation between two persons, but this sentence is not considered by my system. With the application of a system

[5] `https://developer.yahoo.com/geo/geoplanet/`

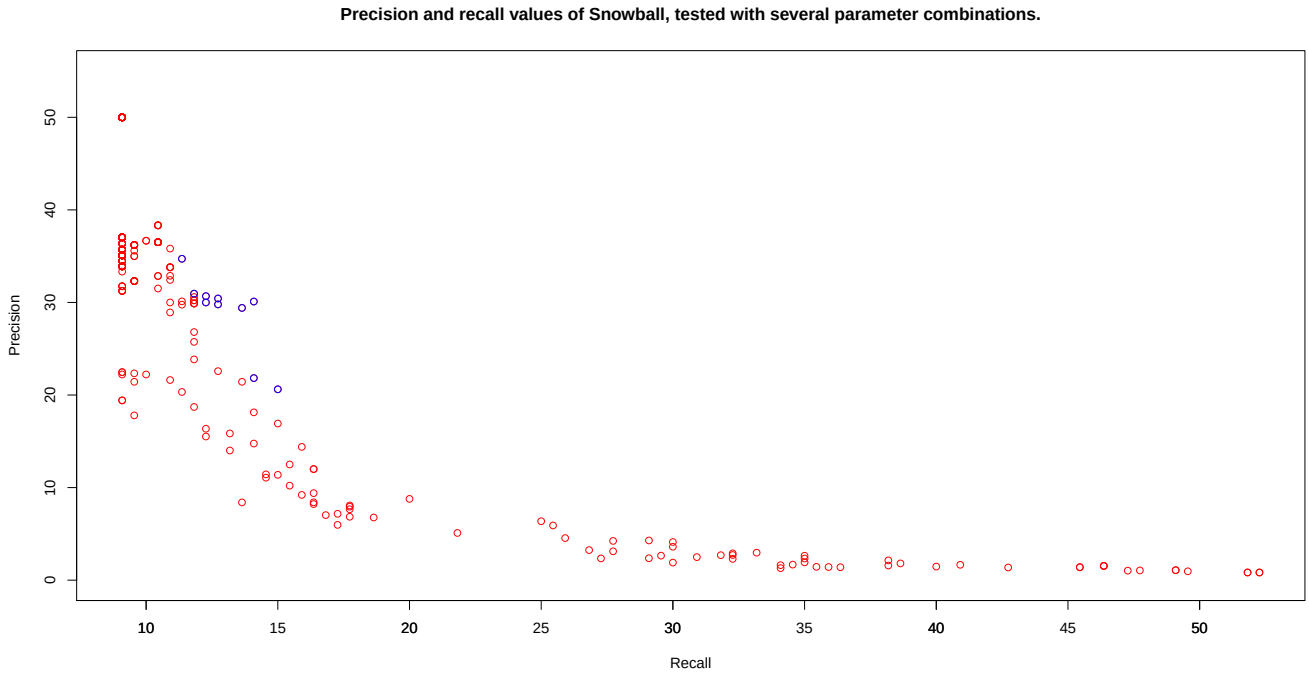

Figure 7: Results obtained by Snowball with various combinations of the parameters.

Book series	Sentences	Locations	Unique location pairs	Extracted relations
Mervyn Peake, The Illustrated Gormenghast Trilogy	24330	56	12	2
China Mieville, Bas-lag Trilogy	49932	266	127	27

Table 5: Tuples extracted from the fiction book.

capable of identifying the person regarding **He** and the person regarding **him**, we would have more sentences to analyse in the relation extraction process, and probably more relationships would be extracted.

6. REFERENCES

[1] E. Agichtein and L. Gravano. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the ACM Conference on Digital Libraries*, 2000.

[2] J. A. Davis. Clustering and structural balance in graphs. *Human Relations*, 20(2), 1967.

[3] J. Diesner, T. L. Frantz, and K. M. Carley. Communication networks from the enron email corpus "it's always about the people. enron is no different". *Computational and Mathematical Organization Theory*, 11(3), 2005.

[4] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Knowledge Discovery in Databases*, 1996.

[5] V. Gardiner. *Changing Geography of the UK: Third Edition*. Taylor & Francis, 1999.

[6] A. Gruzd and C. Haythornthwaite. Automated Discovery and Analysis of Social Networks from Threaded Discussions. In *Proceedings of the International Network of Social Network Analysis Conference*, 2008.

[7] A. Hassan, A. Abu-Jbara, and D. Radev. Extracting signed social networks from text. In *Proceedings of the ACL Workshop on Graph-based Methods for Natural Language Processing*, 2012.

[8] A. Hassan and D. Radev. Identifying text polarity using random walks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2010.

[9] F. Heider. Attitudes and cognitive organization. *Journal of Psychology*, 21, 1946.

[10] M. A. Hernández and S. J. Stolfo. Real-world data is dirty: Data cleansing and the merge/purge problem. *Data Mining and Knowledge Discovery*, 2(1), 1998.

[11] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the International Conference on Machine Learning*, 2001.

[12] S. H. Lee, P.-J. Kim, Y.-Y. Ahn, and H. Jeong. Googling Social Interactions: Web Search Engine Based Social Network Construction. *PLoS ONE*, 5(7), 2010.

[13] A. McCallum, X. Wang, and A. Corrada-Emmanuel. Topic and role discovery in social networks with experiments on Enron and academic email. *Journal of Artificial Intelligence Research*, 30(1), 2007.

[14] Y. Merhav, F. Mesquita, D. Barbosa, W. G. Yee, and O. Frieder. Extracting information networks from the blogosphere. *ACM Transactions on the Web*, 6(3), 2012.

[15] C. Mieville. *The Scar*. Tandem Library, 2002.

[16] C. Mieville. *Perdido Street Station*. Random House Publishing Group, 2003.

[17] C. Mieville. *Iron Council*. Del Rey, 2004.

[18] B. Pang and L. Lee. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 2(1-2), 2007.

[19] M. Peake. *The Illustrated Gormenghast Trilogy*. Vintage, 2011.

[20] L. Ratinov and D. Roth. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Conference on Computational Natural Language Learning*, 2009.

[21] H. Ryu, M. Lease, and N. Woodward. Finding and exploring memes in social media. In *HT*, 2012.

[22] J. Shetty and J. Adibi. Discovering important nodes through graph entropy the case of enron email database. In *Proceedings of the International workshop on Link discovery*, 2005.

[23] J. Turian, L. Ratinov, and Y. Bengio. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2010.

[24] J. Turian, L. Ratinov, Y. Bengio, and D. Roth. A preliminary evaluation of word representations for named-entity recognition. In *Proceedings of the NIPS Workshop on Grammar Induction, Representation of Language and Language Learning*, 2009.

[25] P. D. Turney and M. L. Littman. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems*, 21(4), 2003.

[26] J. Wiebe, R. Bruce, M. Bell, M. Martin, and T. Wilson. A corpus study of evaluative and speculative language. In *Proceedings of the SIGdial Workshop on Discourse and Dialogue*, volume 16, 2001.

[27] L. Zhu. Computational political science literature survey. Technical report, College of Information Sciences and Technology, The Pennsylvania State University, 2010.