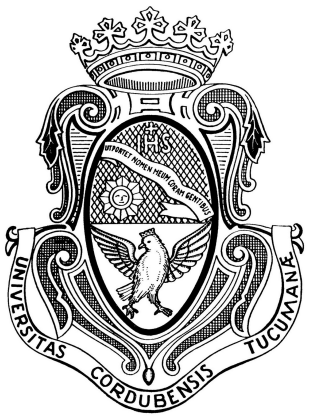
Universidad Nacional de Córdoba

Facultad de Matemática, Astronomía y Física



Graduation Thesis in Computer Science

Function words as a domain independent tool for social role inference

Submitted by: Juan Ignacio Navarro Horñiacek

Supervised by: Luciana Benotti

Collaborators:
Hernan Badenes, Julian Cerruti, Tessa Lau

CÓRDOBA ARGENTINA

Marzo 2013

Function Words as a Domain Independent Tool For Social Role Inference

Abstract

We investigated the automated role inference task in virtual communities under the scope of IBM Communities. We consider this task as a classification problem of two explicit roles: community owners and members as they represent the leader and non-leader roles.

We began with a Human Performance test to measure how well humans can infer leadership roles from the content posted obtaining a 76.25% of accuracy.

Then, we reproduced three standard approaches in Information Retrieval. We trained six automated classifiers to learn which words are relevant for each role so they can be used as domain dependent features for the classification, obtaining 75.26% of accuracy. Finally, our contribution was to propose three domain independent approaches: two of them based in the sociolinguistic theory of politeness, and the third one based in Pennebaker's theory of function words as social carriers. We trained the same six classifiers to test both theories obtaining an accuracy of 69.12% for the Function Words approach, and finding evidence that leaders make use of positive politeness strategies.

We conclude showing that the results achieved by the most common approaches are domain dependent while the domain independent performance of function words is good enough to predict social roles.

Keywords: role inference, *function words*, social interaction, information retrieval, *automated classifiers*, *IBM Communities*.

1 Introduction

In the age of online social networks, most of people's interactions are recorded in virtual communities as they communicate and work everyday. This represents an endless source of data that may help us to understand human social relationships and create tools to automate or assist our daily tasks. The data available online in these communities allow us to extract useful information from them that may be of interest for the identification of roles. In particular, as leadership is present in almost any social group, to identify who is playing leadership roles within a given group becomes an important task in several contexts. But role inference is as complex as language understanding with all of its subtleties. If possible, the applications could be wide enough to be used from expert identification within virtual communities to online leadership in social networks. Therefore, motivated by the possible contribution that represents to identify leadership skills within the members of a virtual community, I focused this thesis in the leadership identification problem.

I performed this research project under a joint research agreement between **IBM** and the **Computer Science Department of the FAMAF - UNC** (Faculty of Mathematics, Astronomy, and Physics of Córdoba's National University) that obtained the SUR Award. The IBM Shared University Research (SUR) Awards is a worldwide award program designed by IBM to promote research in areas of mutual value and interest to IBM and universities.

1.1 Thesis Description

Through this thesis I investigate the automated role inference task in virtual communities aiming to find domain independent features to achieve leadership identification. In particular, I worked within the scope of IBM Communities.

I implemented different approaches in order to compare them and see what techniques work better for the inference task. I also performed a test to measure how well humans can infer leadership roles from the content posted in virtual communities. This gives us an insight about how difficult is the task itself. Finally I present two approaches based in Brown's sociolinguistic politeness theory (Brown & Levinson, 1992) and one based in Pennebaker's psycholinguistic theory saying that function words are carriers of social meaning (Pennebaker, 2011). Comparing these three approaches with the standard methods used in similar works [(Gilbert, 2012), (Diehl *et al.*, 2009), (Diehl *et al.*, 2007)] we present evidence that leaders use positive politeness strategies to communicate effectively and I show that even though a better accuracy can be achieved by these standard approaches, the performance of function words can be good enough.

Four Concrete Goals For this research we defined four concrete goals: The measurement of human performance for the inference task in the scope of virtual communities, the construction of a statistical model to predict the roles within IBM Communities with the highest accuracy as possible, test sociolinguistic and psycholinguistic theories about the use of language in leadership and finally, find domain independent feature that can be relevant for the social role inference.

1.2 Thesis Outline

After this introduction, the main motivations and related work is introduced in Sections 2, 3 and 4 where the goals of this research are established as we present our definition of the problem. Afterwards, in Sections 5 to 7 we describe the data available in IBM communities and we analyze it aiming to find a linguistic relation between the defined roles and the words used in the content of the posts. We also present the human performance for the role inference task as a gold standard, and then we describe each of the approaches proposed to use later in the automated classification task, with the construction of the datasets for each one of them. In Section 8 we present the results of the automated classification process and we analyze the accuracy obtained by each one of the selected classifiers with a deeper insight in our best classifier's results in Section

9. Finally, as a matter of conclusion, in Section 10, we present our contribution showing that function words are a robust alternative for the role inference based on language usage and we finish with several ideas for future work.

2 Related Work

In the last years, as social networks became tools used daily by several users, scientists started working to develop techniques to achieve automated role inference as they realize of the importance of this task. There are different approaches to it, like the one published by Gilbert about detecting hierarchies automatically within a given company (Gilbert, 2012). He took the jobs titles, and the email corpus from a company and established a rank of positions and then based on a machine learning approach, using n-grams found in those email interactions, he found phrases that reveals 'upwards', 'downwards' and 'equal' relationships.

Another approach made by Weerkamp and de Rijke was to define credibility indicators (Weerkamp & Rijke, 2012) for finding expert solutions in QA forums, assuming that experts communicates in such a way that inspires credibility from their readers. Diehl, Montemayor, and Pekala made a similar effort defining social attributes for people instead of the typical repositories of unrelated information (Diehl *et al.*, 2009), allowing us to search within a social network more efficiently by social queries. An example of this would be to look for someone who is similar to one's best friend, instead of looking for someone who has the same interests in their profiles as we usually search in information repositories.

Diehl have also attempted with Namata and Getoor to model social relationships as interactions (Diehl *et al.*, 2007), and then ranking each of these interactions as relevant or not according to the type of relationship is intended to look for. In particular they aimed to identify a manager-subordinate relationship using a corpus of emails from a company. Based on the hypothesis that a manager will share messages with several subordinates and vice-versa, they looked at the traffic of the messages and their direction, and also they looked at the term frequency to focus in the terms that are highly related to this manager-subordinate relationship to determine which interactions were relevant.

Prabhakaran studied how different power relations are manifested in the structure and language of online written dialogs and built a system to automatically extract power relations from them (Prabhakaran *et al.*, 2012). Then, himself, Rambow and Diab (Prabhakaran, 2012) performed a theoretical analysis of the language used in specific power relationships (hierarchical relations within a company), and analyzes them with an automatic system. They looked for overt display of power (ODP) in communication interactions, saying that an ODP is found when an utterance is interpreted as creating additional constraints on the response beyond those imposed by the general dialog act. Examples of this can be found in the following sentences: 'I need the answer ASAP' and 'Please give me your views', while it is not present in the following ones: 'Enjoy the rest of your week!' or 'would you agree that the same law firm advise on that issue as well?'. They had manually annotated a corpus of emails from a company, and

then trained a tagger to perform the task over unseen utterances. So their work consisted basically in classifying automatically each utterance as having or not an ODP.

Our main difference is that we intended to build domain independent models that are able to generalize its application to several different contexts by introducing approaches that uses domain independent features to achieve the social role inference. Also our work is different in that we only use unigrams for this task and that we tried more automated classifiers in order to cover different techniques for the learning process.

3 Definition of Domain Independent Features

We would like to obtain results that are able to generalize to several different contexts, meaning datasets of different domains or topics, that shares a common structure of the information but they differ in their content: the terms of phrases used by the authors. It is clear that datasets from different domains will not share the same characteristics, so to generalize our results we want to find features that are present in most of the contexts. Therefore we call **domain independent features** to those characteristics that are present in different contexts.

For our particular dataset obtained from the IBM Communities databases, we had the advantage of having annotated the roles by IBM according to the structure of the leadership team stored at the moment of adding each of the users to the communities, but this is not usual in most of the forums and virtual communities in the internet, and moreover it is very difficult to annotate them manually. Therefore, if we are able to build statistical models that also works for other communities, i.e. domain independent models, then we will not need to manually annotate this roles in the target communities where we seek to infer the roles.

4 Definition of the Problem

We based this research on the IBM communities, an internal social network within IBM that stores forums, wikis, blogs, files and bookmarks grouped by interests or topics. This communities are mainly used for knowledge sharing amongst members of the company, social interaction as to build each others reputation inside IBM, collaborative team work and the use of communication as a vehicle for the leadership team to the members. We consider the role inference task under the scope of these virtual communities as a classification problem of the two roles within them: Members and Owners.

Owners are those who are in charge of the organization and leadership of the community and sometimes also includes experts/managers considered as useful contributors. Members as their name indicates are common users that participate in the communities, but are not recognized as playing a leadership role. We studied members and owners not only because automatically recognizing them could enable personalized tools for the different roles to help care for their

community better, but because these roles represent the leader and non-leader roles in the most common scenarios of virtual communities or discussion forums.

The IBM Communities dataset has explicit role labels for each user within a community, therefore we can use it as a gold standard to calculate the accuracy for each prediction, and there is no need to manually annotate or guess each user's role. They can be owner users or member users as it is described below.

What is an Owner? An owner is a user that has an ownership role within a virtual community. Owner users have a special use of the community, and we have the hypothesis that it is revealed in their use of language. Owners' role is to maintain and lead the community so we believe that they write in a different way than members do. Generally they address their speech to everyone in the community, they announce decisions, explain solutions, organize events, ask for feedback from the entire group of users, and motivate all the users in the community to interact with each other. They are usually managers, team leaders, project managers, or experts in a specific field within IBM and for these reasons we believe they are more polite than members in the way they write.

What is a Member? Member are the basic users of a community, they participate in several ways, like looking for answers to their questions, or participating as they were requested. They commonly talk about themselves and their problems and ask questions about how to solve them. Even when sometimes they act very similar to an owner user contributing to the community, sharing knowledge, and acting like leaders we believe that they are more informal in their language like using contractions and having spelling errors. We also expect them to use more technical terms.

5 Characterizing the Data

An IBM community has 7 kinds of publications inside them: Forum posts, Forum replies, Blogs, Blog comments, Files, Wikis and Bookmarks. We were given access to 194 public communities, with 8055 authors, 54963 entries in total, and the following proportion for each kind of entries: Forum replies (28.72%), Bookmarks (19.31%), Blog entries (14.73%), Wiki entries (11.35%), Blog comments (9.18%), Forum posts (9%), Files (7.65%). As we are working with the English language, it is also important to note the proportion of the authors per country just to take into account that for many authors English is not their native language. The 5 countries with the most participation in these communities are: USA (44.47%), Brazil (11.35%), India (7.19%), China (6.65%) and Canada (4.41%). Another interesting thing to note is that 22.21% of the users have manager positions within IBM while 77.79% don't. From this 8055 authors, 95.28% are member users and 8.27% are owner users in all the communities in which they participate, while the remaining 3.55% are users that have an owner role in some communities while they are members in other communities.

We decided to focus in Forum entries because they are mainly being used for brainstorming and discussions, discarding forum replies by now because they

typically contain short answers to questions or very short comments. We also discarded Files, Bookmarks and Wikis because they don't have conversations, or any other language interactions between users. Finally, we also decided not to include Blogs and Blog comments, because usually they are large, containing rich text and images, and most of the time they are not oriented to discussion or interaction between users but being a mere publishing place.

Table 1. Basic Analysis of owner vs member forum posts in IBM Communities.

Owner Forum Posts	Member Forum Posts
Amount of posts: 1733	Amount of posts: 3221
Words used in total: 301227	Words used in total: 398651
Average words per post: 173.8	Average words per post: 123.8
Vocabulary size: 8371	Vocabulary size: 15526
Normalized vocabulary size: 4.83 words/posts	Normalized vocabulary size: 4.82 words/posts

The Forum post corpus has a size of 4954 posts, made up of 699878 words, having an average of 141.3 words per post. This corpus contains 1733 Owner posts, using 301227 words in total, with an average of 173.8 words per post, and 3221 Member posts, using 398651 words in total, with an average of 123.8 words per post. As a summary of the obtained results we can observe in Table 1 that the vocabulary size (the amount of used words by each role without repetitions) for members is twice greater than the vocabulary size for owners, but also the amount of member forum posts is twice than owner forum posts. Their normalized vocabulary (the vocabulary size divided by the amount of posts) is practically the same, meaning that they had a similar rate of addition of new words per post. We also observed that there are different pronouns or keywords being used more frequently in one role than in the other. The following list shows the first 25 more frequent words in descending order for both roles.

Member's 25 most frequent words:

the	in	on	be	can
to	i	that	we	as
and	for	this	you	or
a	is	it	are	not
of	ibm	with	have	will

Owner's 25 most frequent words:

the	in	on	it	or
to	for	that	we	will
and	is	this	be	i
of	you	with	as	have
a	ibm	are	your	can

It is interesting to note that members have a higher frequency for words like: 'I', 'my', 'some', 'help', 'thanks', etc, while owners have a higher frequency for words like: 'you', 'we', 'team', 'please', 'community', etc.

6 Human Performance Test

It is well known that in Natural Language Processing humans have better performance than machines, so we would like to set it as an upper bound, and then try to imitate the human role inference performance with an automated approach.

In order to do this, a representative sample of 120 owner and member post's contents was taken from our dataset to perform a poll so we can ask two of our collaborators to act as annotators and classify each post of the sample as something written by a member user or by an owner user. After they completed the poll, their answers were compared to the database record of the actual role for each post's author as a gold standard, in order to calculate their accuracies.

Both annotators obtained an accuracy of 75% and 77.5% respectively, but it is important to note that even when the annotator with the highest accuracy was an specialist working in IBM for more than 10 years, both accuracies are not significantly different even with a low level of confidence such as 80% getting a p-value of 0.6503. This means that there is no significant difference for humans in being an expert or not in the specific domain when it comes to infer roles, showing evidence that in fact humans uses domain independent features for this task. For practical purposes, in order to compare with the automated classifier's results we can average both human accuracies and claim to have 76.25% of accuracy for the Human Performance Test. We also obtained a κ value (Cohen's Kappa coefficient), an inter-rater agreement measure, of 0.682 what is considered as 'substantial' in Landis and Koch scale (Landis & Koch, 1977). This validates their classification, and confirms that the task is not trivial even for human beings.

7 Feature Selection

We decided to work in this project only with post level features in order to give a more linguistic approach to the problem. To select the features, we implemented basic strategies with the classic domain dependent approaches first, and then we introduce three domain independent approaches: Pronouns, Verbs vs Nouns and Function Words to test sociolinguistic and psycholinguistic theories of language.

In order to compare the new approaches, we reproduced Gilbert's (Gilbert, 2012) approach, reducing the domain specific words by removing outliers (uncommon words) and using the remaining words as a bag of words approach to learn which words are predictive for each role. Then, we tried to enhance this first approach by stemming the words first and then applying the same process of removing the outliers. We were motivated after finding different conjugations of the same word, thinking that stemming may map them to the same token increasing the weight of each predictive word. We also tried removing the function

words and the outlier words from the dataset in order to compare the accuracies with the first two approaches, and evaluate if this approach, lacking of function words, can obtain the same accuracies than with function words and if it becomes even more domain dependent. When exploring the data to find relevant features, it was evident that pronouns could be an important predictor for these roles, and our main motivation to use pronouns was to prove one hypothesis of politeness theory which claims that when it is important to maintain a close relationship a positive politeness strategy is to use more inclusive and 2nd person pronouns. So we analyzed looking for evidence to validate this hypothesis. We also observed presence of polite language in owner posts and this motivated us to prove one hypothesis of politeness theory which claims that when there is a social difference or hierarchy a negative politeness strategy is to use more nouns than verbs in the sentences in order to take distance from the action. So we implemented the Verbs vs Nouns approach. Finally, we were motivated to prove Pennebaker's (Pennebaker, 2011) theory which claims that function words are carriers of social meaning and thus, they reveal the underlying social relationships between the agents of a communication process, by implementing our Function Words approach.

We understand these approaches as filters that we can apply to the post's contents in order to set up different datasets. Each of these approaches was instantiated with automatic classification algorithms as it is described in Section 8 and 9. A more detailed description of each approach and the definition of their datasets can be found in Appendix A, a detailed description of the results can be found in Appendic C.

8 Classifiers' Results over each dataset

As we can observe in Table 2, the best classifier for all the datasets was Simple Logistic. In second and third place for OR, S_OR and OFR datasets the classifier's with the highest results were SMO and J48 respectively, it is important to note here that the results obtained by them are not significantly different from the Human Performance accuracy. In Pron dataset the best classifiers were Simple Logistic, J48 and Naive Bayes, but for the rest of the datasets Naive Bayes was always below the Majority Class baseline. For VsNs dataset the best classifiers were Simple Logistic, Logistic Regression and J48 respectively, but the results obtained by all the classifiers for this dataset were not significantly different from the Majority Class baseline. In FW dataset, the best classifiers were Simple Logistic, Logistic Regression and SMO respectively, obtaining for the three of them a similar accuracy that represents near the half of improvement from the Majority Class baseline to the Human Performance accuracy. A detailed description of the classifiers is given in Appendix B.

These results show several things, in first place that the OR, S_OR and OFR datasets are obtaining the same accuracy as in the Human Performance fulfilling this way our second goal for this project as seen in Table 3. In second place, we can see that the two best results on the Pron dataset are significantly differ-

Table 2. Accuracies obtained by all classifiers over all the datasets

Classifier Name	OR	S_OR	ORF	Pron	Vs-Ns	FW
Simple Logistic	**75.26%**	75.05%	74.2%	66.92%	65.72%	69.12%
SMO	**74.64%**	73.9%	73.42%	65.81%	65.02%	69.04%
J48	**72.38%**	71.56%	69.8%	66.9%	65.16%	66.71%
OneR	67.88%	**68.17%**	67.89%	65.54%	61.77%	65.62%
Majority Class	65.37%	65.37%	65.39%	65.02%	65.02%	65.02%
Naive Bayes	65.35%	64.9%	65.02%	**66.15%**	64.53%	64.03%
Logistic Regression	61.82%	65.56%	64.11%	66.67%	65.34%	**69.1%**

* The best result for each classifier is in bold. The best results for each dataset is in green. The worst result in red and the majority class baseline is in blue.

ent from the Majority Class baseline with 95% of confidence as seen in table 4, showing the evidence that leaders tends to use this positive politeness strategy of using inclusive pronouns to communicate effectively. In a similar way, looking into the VsNs dataset results, we can say that we don't have enough evidence to believe that leaders use this negative politeness strategy to communicate with the community. Lastly, FW dataset obtained a substantial result for three of the classifiers, proving that function words are good predictors for social roles as stated in Pennebaker's theory. As we can see in the table 4 where the significantly different accuracies are in bold text, it is very interesting to note that even when Pron and VsNs results were significantly different from the Human Performance, the FW dataset and Human Performance accuracies were not significantly different with a p-value of 0.0716. However, the difference could have been statistically significant if a bigger sample was used in the Human Performance test. A better description of the results for each approach can be found in the Appendix.

Table 3. Statistical Significance in Bold (95% of confidence) - Domain Dependent.

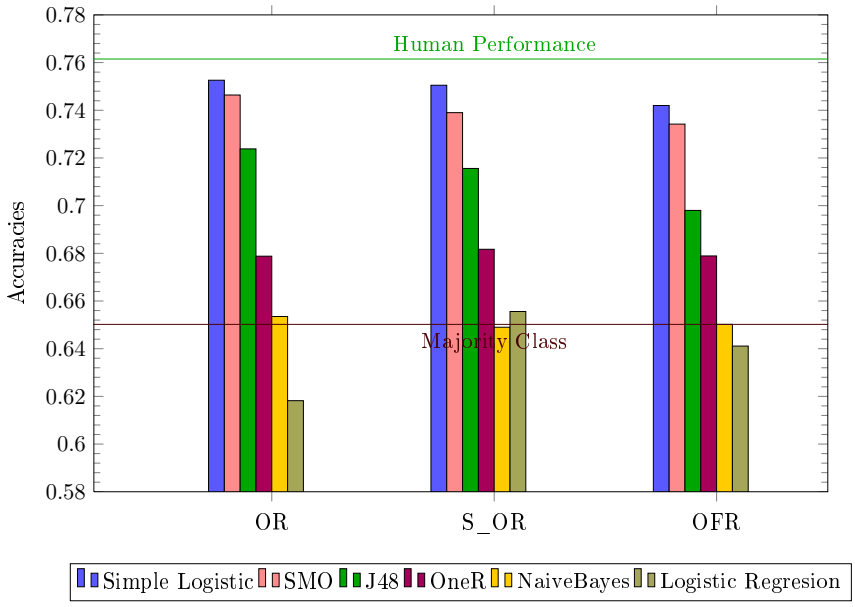
Dataset	OR	Size	Human Performance	Size	P-value
Simple Logistic	75.26%	4866	76.25%	120	0.8041
SMO	74.64%	4866	76.25%	120	0.6855
J48	72.38%	4866	76.25%	120	0.3289
Dataset	S_OR	Size	Human Performance	Size	P-value
Simple Logistic	75.05%	4866	76.25%	120	0.7613
SMO	73.9%	4866	76.25%	120	0.552
J48	71.56%	4866	76.25%	120	0.2356
Dataset	OFR	Size	Human Performance	Size	P-value
Simple Logistic	74.2%	4865	76.25%	120	0.6039
SMO	73.42%	4865	76.25%	120	0.4739
J48	69.8%	4865	76.25%	120	0.103

Table 4. Statistical Significance in Bold (95% of confidence) - Domain Independent.

Dataset	Pron	Size	Majority Class	Size	P-value
Simple Logistic	**66.92%**	4954	**65.02%**	4954	0.0459
J48	**66.9%**	4954	**65.02%**	4954	0.0483
Logistic Regression	66.67%	4954	65.02%	4954	0.0833
Dataset	VsNs	Size	Majority Class	Size	P-value
Simple Logistic	65.72%	4954	65.02%	4954	0.4641
J48	65.34%	4954	65.02%	4954	0.7382
Logistic Regression	65.16%	4954	65.02%	4954	0.8838
Dataset	FW	Size	Majority Class	Size	P-value
Simple Logistic	**69.12%**	4954	**65.02%**	4954	0
Logistic Regression	**69.1%**	4954	**65.02%**	4954	0
SMO	**69.04%**	4954	**65.02%**	4954	0
Dataset	FW	Size	Human Performance	Size	P-value
Simple Logistic	69.12%	4954	76.25%	120	0.0716
Dataset	Pron	Size	Human Performance	Size	P-value
Simple Logistic	**66.92%**	4954	**76.25%**	120	0.0184
Dataset	VsNs	Size	Human Performance	Size	P-value
Simple Logistic	**65.72%**	4954	**76.25%**	120	0.0078

8.1 Results for the 3 domain dependent approaches

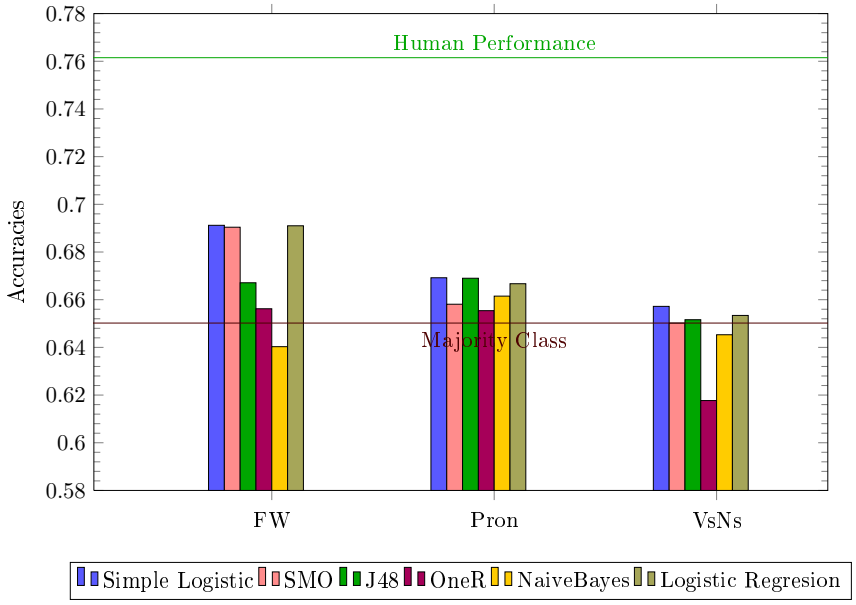
Figure 1. Accuracies of the Domain Dependent Approaches



In Figure 1 we can see that the OR dataset obtained the highest accuracy being statistically the same value as the obtained in the Human Performance test, specially for Simple Logistic, SMO and J48, but we also can see that it obtained the lowest accuracy for Logistic Regression classifier. We can also see that S_OR dataset had more near results between all the classifiers, but their accuracies decreased respect to the OR dataset. In a similar way the accuracies obtained by the OFR dataset decreased even more respect to the OR and S_OR datasets' results, contrary to what is expected in most of the NLP and Information Retrieval tasks.

8.2 Results for the 3 domain independent approaches

Figure 2. Accuracies of the Domain Independent Approaches



It is clear in Figure 2 that the NsVs dataset isn't significantly different to the Majority Class baseline for its best classifiers' results. Also it is easy to see that Pron dataset obtained an improvement over the Majority Class baseline specially with Simple Logistic and J48 classifiers showing evidence that leaders uses inclusive and 2nd person pronouns to communicate. Finally we can see that FW dataset obtained a considerable accuracy for three of the classifiers (Simple Logistic, SMO and Logistic Regression) showing that function words are good domain independent predictors for the social role inference task.

9 Simple Logistic: A Detailed Analysis of Our Best Classifier

In this section we intend to perform a special analysis to the results obtained by the Simple Logistic classifier for the OR, S_OR, OFR and the FW datasets in order to show that the high accuracies obtained by the OR, S_OR and OFR datasets is because most of the words that have been weighted as relevant are domain dependent and are working well only for this given context.

Taking into account how this classifier works, basically building two functions to calculate a value of 'ownerness' and 'memberness' for each posts respectively and then taking the higher value obtained by both functions as the prediction, we would like to see which words are weighted as more relevant for each role. This output would be useful to validate if the words expected to be relevant were also considered as relevant by the classifier and eventually, it can also be used as feedback to redefine new features for role inference. In other words, the prediction is given by the following algorithm: $Max\ (C_{member}(post), C_{owner}(post))$

where: $C_{(member|owner)}(post\,content) = \sum w_i * F_i$ and where w_i is the weight of the word (feature) F_i in the dataset and $C(p)$ is the caculated value for post p. First, for all the observed words in the OR, S_OR, OFR and FW datasets we defined the following categories that contain them:

I - Community Related words
II - Date and Time
III - Manager/Team-Leader words
IV - Informal/Follower words
V - Technical words
VI - Domain Specific words
VII - Neutral words

Then we had a human annotator to tag each of the words as belonging to these categories listed above. Afterwards, without looking to the output of the classifiers, we decided that words of categories I, II and III are more expected to be used by leaders, while words of the category IV are more expected to be used by non-leaders. Words that belong to the categories V and VI are considered as domain dependent, and therefore not relevant for the inference of a leadership role in a more general context. So then we took from the functions that predicts owner role only the words with positive weights. In other words, $\forall i\{F_i|w_i > 0\}$, and considered them as the 'owner words'. And then we did the same for the function that predicts member role obtaining the 'member words'.

After we made up this two separated lists of words, we tagged them as relevant or as domain dependent. For owner words we tagged them as relevant if they belong to categories I, II or III, and for member words if they belong to category IV. We tagged owner and member words as domain dependent if they belong to categories V or VI. After tagging all the member and owner words, we calculate the proportion of relevant and domain specific words on each dataset. Obtaining the following results:

Table 5. Proportion of Relevant and Domain Specific words for each dataset in the Simple Logistic Classifier's Output

Type of words	OR	S_OR	ORF	FW
Owner relevant words	40.6%	42.8%	40%	36.1%
Owner domain dependent words	21.8%	23.8%	22.8%	0%
Member relevant words	18.3%	22.2%	15%	29.6%
Member domain dependent words	32.4%	18.5%	30.2%	0%

OR dataset obtained the highest accuracy, but as we can observe in Table 5, they have a high proportion of domain dependent words in the prediction of both roles, in particular it has the higher proportion of domain dependent words for Member role prediction. This makes us think that its prediction is getting very good results thanks to the domain specific words. It is important to consider that function words are included within OR dataset (they pass all the filters), and also that member prediction is better for each classifier.

S_OR dataset has the highest proportion of relevant words found for owners, and this might be happening because stemming maps several conjugations of the same word to a unique stem increasing their relevance for the classifier, and thus, their weight in the output. On the other hand, this approach also has the highest amount of domain specific words for owner words list. Obviously, domain specific words usually are nouns, so they will remain as they are after the stemming process. Having more weight in domain specific words and more weight in relevant words, fewer words are needed in the function given as output for the prediction of each role.

The **OFR dataset** obtained a similar accuracy than the OR dataset generally increasing the domain specific words and decreasing the relevant words for owner words list. It is also interesting that it gets the lowest proportion of relevant words for member words. So clearly is getting a high accuracy by increasing the domain specific words.

Finally, the **FW dataset**, with no domain specific words at all, has the highest proportion of member relevant words. Also, If we consider that function words are inside OR dataset, and looking how OFR dataset increases domain specific words to maintain its accuracy, we can follow that function words are supporting considerably the OR dataset results. We have evidence to believe that the statistical model built from OR dataset won't obtain a similar accuracy in post contents from a different domain, but we do expect for the FW dataset to maintain its accuracy over different domains.

In summary, the accuracy we are getting with the OR dataset of 75.26%, being almost 10% over the Majority Class baseline may be domain dependent because we observe big proportions of domain specific words in their statistical models. Therefore we cannot guarantee that the 75% of accuracy will be maintained if we moved to another domain (e.g. discussion forums in a marketing company). However, function words on their own are reaching a considerable 69.12% of accuracy and we expect this result to actually be domain independent because all domains of discussion use them. Therefore we can propose Function

Words as the domain independent features we were looking for in order to use in the role inference task, fulfilling our last goal for this project.

10 Conclusion and Future Work

We began making a Human Performance test to measure how well humans can infer leadership roles obtaining a 76.25% of accuracy achieving our first goal. Afterwards, using only the forum posts from the IBM Communities we reproduced three basic approaches in Information Retrieval. We trained six automated classifiers to learn which words are relevant for each role. The results obtained, 75.26% of accuracy for our best classifier, can be interpreted as the same value obtained in the Human Performance test because their difference is not statistically significant. In other words, we achieved our second goal as we obtained the Human Performance accuracy with automated classifiers.

Then, we present our main contribution proposing three domain independent approaches based in the theories we wanted to prove. Two of them based in Brown's and Levinson's politeness theory, and the third one based in Pennebaker's theory of function words as social carriers. Training the same six classifiers, we built our statistical models to learn which words are related to each role. We obtained a best accuracy of 69.12% for Function Words, 66.92% for the Inclusive and 2nd person Pronouns, and 65.72% for Verbs vs Nouns, showing evidence that leaders use positive politeness strategies to communicate effectively.

Finally, analyzing the results we conclude that function words performance is good enough to predict social roles and we expect this result to actually be domain independent because function words are used as the 'connective tissue' of language. Thus, we demonstrate that Function Words are the domain independent features we were seeking for the role inference task. This contributes to a shift of paradigm seeing function words, not as a mere syntactic element anymore but, as a pragmatic component or a tool for social meaning.

For Future work, we will implement another approach to the problem by classifying authors instead of posts. We will also try training the classifiers with fewer training posts as we suspect that function words' accuracy will remain while the other approaches' accuracies will decrease. On the other hand, going into deep NLP by annotating with humans the part of speech tags for a sample, and then training a tagger or parser over those annotated posts can lead us to better results for the negative politeness approach. Finally, an interesting experiment will be to use the already obtained statistical models to predict the roles for each community separately.

References

Brown, Penelope, & Levinson, Steve. 1992. *Politeness: Some Universals in Language Usage - Studies in Interactional Sociolinguistics.* Vol. 4. Cambridge University Press.

Diehl, Christopher P., Namata, Galileo, & Getoor, Lise. 2007. Relationship identification for social network discovery. *Pages 546–552 of: Proceedings of the 22nd national conference on Artificial intelligence - Volume 1*. AAAI'07. AAAI Press.

Diehl, Christopher P., Montemayor, Jaime, & Pekala, Mike. 2009. Social Relationship Identification: An Example of Social Query. *Pages 381–388 of: Proceedings of the 2009 International Conference on Computational Science and Engineering - Volume 04*. CSE '09. Washington, DC, USA: IEEE Computer Society.

Gilbert, Eric. 2012. Phrases that signal workplace hierarchy. *Pages 1037–1046 of: Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*. CSCW '12. New York, NY, USA: ACM.

Landis, J. R., & Koch, G. G. 1977. The measurement of observer agreement for categorical data. *Biometrics*, **33**(1), 159–174.

Malecha, G., & Smith, I. 2010. *Maximum Entropy Part-of-Speech Tagging in NLTK*. unpublished course-related report: http://www.people.fas.harvard.edu/ gmalecha/.

Pennebaker, James W. 2011. *The Secret Life of Pronouns: What Our Words Say About Us*. Bloomsbury USA.

Porter, M. 1980. An Algorithm for Suffix Stripping. *Program*, **14**(3), 130–137.

Prabhakaran, Vinodkumar. 2012. Detecting power relations from written dialog. *Pages 7–12 of: Proceedings of ACL 2012 Student Research Workshop*. ACL '12. Stroudsburg, PA, USA: Association for Computational Linguistics.

Prabhakaran, Vinodkumar, Rambow, Owen, & Diab, Mona. 2012. Predicting overt display of power in written dialogs. *Pages 518–522 of: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. NAACL HLT '12. Stroudsburg, PA, USA: Association for Computational Linguistics.

Weerkamp, Wouter, & Rijke, Maarten. 2012. Credibility-inspired ranking for blog post retrieval. *Information Retrieval*, **15**(3-4), 243–277.

Witten, I.H., & Frank, E. 2005. *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition*. Data Mining, the Morgan Kaufmann Ser. in Data Management Systems Series. Elsevier Science.

Appendix

A A Detailed Description of Each Approach

A.1 Removing Outliers Approach

Based on Gilbert's work (Gilbert, 2012) introduced in Section 2, we designed filters to run over the dataset in order to discard all these domain-specific words because of our broader goals of creating a domain independent solution.

The applied filters were the following:

IBM Communities-dependent: Our goal was to remove outliers because we assume they are specific to this dataset. The first filter to apply should be simple and should reduce the amount of words in a way that we end up with a bag of words that can be easily handled. In first place we should remove all those words that were misspelled or that were used only few times within the dataset. So, the first filter was to discard all the words that didn't appear at least 9 times in all the dataset. After doing this we end up with only a lexicon of 4677 words.

Author-dependent: We know that each person is different in the way they write, there are some people who write in a very particular way, and giving this to a classifier would mislead the training process. The second filter we applied was to discard all the words that were not written by at least 3 different authors. After applying this filter the lexicon was reduced to 4546 words.

Post-dependent: It might be possible that there are some words that appeared several times in different posts because it was necessary in order to explain something specific or share some information a couple of times within a community, so the third filter we decided to apply was to discard all the words that didn't appear in at least 10 different posts. After this we had a lexicon of 4045 words.

Community-dependent: We also realized that there are some domain specific words related to the topic of a community, that can be found naturally several times. In order to solve this problem, we applied another filter discarding all the words that didn't appear in at least 5 different communities. Finally we obtained a lexicon of 3917 words, discarding 128 words: 50 portuguese words, 61 domain specific words, and just 17 general/neutral words.

We call **OR dataset** (Outliers Removed dataset) to the results of applying the filters described above to the IBM Communities dataset.

A.2 Enhancing through Stemming Approach

When we look the entries as we performed the human performance test, we observed that there are words with the same root but different conjugation used several times. For example for the words: 'replying', 'reply', 'replies'.

We hypothesize that mapping them into the same token it will improve the relevance for a word meaning on each role. The two options we considered were stems or lemmas. Stemming is a process of removing and replacing word suffixes to arrive at a common root form of the word. Lemmatization consists in using a dictionary to lookup lemmas for each word. Lemmas differ from stems in that a lemma is a canonical form of the word, while a stem may not be a real word. For example, the stem of 'replying', 'reply' and 'replies' is 'repl' and it isn't a real word in English language.

Usually if the language has a rich morphology and many irregularities (eg. Spanish or French), we would prefer lemmatization because words will have very different roots or stems on each of their conjugations found in the dataset. This is not our case because the English language is not that morphologically rich and most of their words share the same root for their conjugations, except for the irregular verbs. In particular, as we have several words that are domain specific to IBM communities, like company's department names, project names, name of tools or software packages, or even just words created by the users of the communities, with presence of spelling errors and colloquial language, we can expect this highly noisy data to have a considerable percentage of out-of-dictionary words, so even if we use lemmatization, an analysis of this percentage should be done.

Based on these reasons, we decided then to use the stem of the words to do it and we used the Porter Stemmer algorithm (Porter, 1980) provided by NLTK package.

We run the stemmer over the IBM Communities dataset first, and then we applied the same filters described in section 3.3.4 (dataset dependency, author dependency, post dependency and community dependency), thus we will call this dataset **S_OR dataset** (Stemmed -> Outliers Removed dataset). After filtering this dataset the lexicon was reduced to 2886 words.

A.3 Removing Outliers and Function Words Approach

We were aware that function words are included within OR dataset as they passed all the filters proposed for that approach, therefore it would also be interesting to see how the OR approach works without including Function Words and then compare their results.

After we built the OR dataset, we filter the same Function Words we used in Section A.6, and we define a new dataset called **OFR dataset** (Outliers and Function Words Removed dataset). Our hypothesis is that this dataset will get a lower accuracy than the OR dataset. We also expect that the amount of domain specific words selected as relevant by the classifiers will increase trying to maintain the accuracy achieved by OR dataset.

A.4 Pronouns Approach

Based on what we saw in the posts during the human performance test, we developed a general idea about what kind of words were related to each role. So the first step to take was to validate it with a simple and easy-to-implement method. We started looking at the frequency of words related to each role. Thus, we decided to define the following two sets of words:

2nd person pronouns: you, your, we, us, our
1st person/indefinite pronouns: I, my, me, mine, anybody, somebody

We filtered all the words in order to discard those terms that were not in any of these two sets, and then we calculated the proportion of member words and owner words on each post. Based on the output we also calculated the average of the words that belongs to each set for member and owner posts separately.

Table 6 shows the average frequency of the words related to each role calculated in both owner and member forum posts. What we can see at first glance is that owners used a higher proportion of 2nd person pronouns than 1st person/indefinite pronouns. In contrast, members used an equal proportion of 1st person/indefinite pronouns and 2nd person pronouns.

Table 6. Average of words per post

Averages	2nd person pronouns	1st person/indefinite pronouns
Owner posts	0.028	0.007
Member posts	0.017	0.020

It is expected that owners use more 2nd person pronouns when they write, and it seems that they write uniformly between owners, therefore we had a higher frequency of 2nd person pronouns when looking at owner posts. Usually there is an expected behaviour for an owner, there are specific rules that owners have to follow, they even get training sessions on leadership skills.

On the other hand, it makes sense that the results were not so predictive when looking at member posts because there are different kind of users, and some of them have (maybe unknown by themselves) leadership skills even when they are not officially leaders, managers or owners of the community. So they don't write in a uniform way, they use 1st person/indefinite pronouns and 2nd person pronouns equally.

In conclusion, these sets of words related to each role could confirm us that we were moving in the right direction, and give us a basic method to predict each user's role. Motivated by this first analysis and by the claim of possitive politeness which says that there is a greater use of 2nd person pronouns and inclusive pronouns when it is intended to mantain a closer relationship, we implemented this approach. We call this the **Pron dataset** (Pronouns dataset).

The words used to filter the dataset were the following words:

we, my, anybody, you, our, your, i, own, somebody, mine

A.5 Verbs vs Nouns Approach

One of the hypothesis introduced in Negative Politeness says that the more nouns there are in an expression, the more formal or polite it becomes (Brown & Levinson, 1992). When you use more nouns you are removing the actor from doing or feeling or being something. The actor becomes an attribute (e.g. adjective) of the action. Let's look at a simple example:

a) You performed well on the examinations and impressed us.
b) Your good performance on the examinations impressed us.
c) Your good performance on the examinations was impressive to us
d) Your good performance on the examinations made a good impression on us.

As the verbs become nouns from a) to d) we get more formal sentences. We hypothesized that owners are more polite and we wanted to test if they will have a higher noun/verb ratio than members in their sentences. Therefore, one of the implemented features was the relation between nouns and verbs.

To add the part of speech tags we used the the Maxent Treebank POS tagger (Malecha & Smith, 2010) trained on Penn dataset provided by the Python NLTK package. To construct this dataset we discarded the words including only the tags in order to take them as features to train the classifiers in the relation between verbs and nouns.

There were different categories for verbs and nouns, so we also included them separately because they may contribute more information to the classifier. For example, it is more common to have verbs in the past participle tense in polite sentences (because of their use in passive voice) than verbs in the simple present tense.

The verbs categories are:

- modal verbs (md)
- verbs in the simple present tense (not third person) (vbp)
- verbs in the gerund tense (vbg)
- verbs in the simple past tense (vbd)
- verbs in the past participle tense (vbn)
- verbs in the infinitive tense (vb)
- verbs in the third person (she, he, it) of the simple present tense (vbz)

The nouns categories are:

- singular nouns (nn)
- plural nouns (nns)
- proper nouns (nnp)

As we only wanted to look at verbs and nouns we made a dataset with only their proportions as a vector of numeric features and we called this dataset **Vs-Ns dataset** (Verbs vs Nouns dataset).

A.6 Function Words Approach

Function words are commonly defined as natural language words which are very frequent, playing a syntactic role and are considered to have a lack of semantics, such as 'and', 'the', 'a', 'an', and similar terms. Therefore, they are usually excluded when performing information retrieval tasks.

Function words are words that have little lexical meaning or have ambiguous meaning, but instead serve to express grammatical relationships with other words within a sentence, or specify the attitude or mood of the speaker. They contribute to the structural relationships that words have to one another. Thus, they serve as important elements to the structures of sentences. Each function word gives some grammatical information on other words in a sentence or clause, and cannot be isolated. Usually the understanding of function words have been always as a syntactic component, but if it is possible to infer roles with them, we will be able to believe they are also social carriers as Pennebaker claims in his book The Secret life of Pronouns. (Pennebaker, 2011) There are four major open classes of words that occur in the languages of the world: nouns, verbs, adjectives, and adverbs, but Function words don't belong to any open class. They are mainly: articles, pronouns, adpositions, conjunctions, auxiliary verbs, interjections, particles, expletives, pro-sentences, prepositions, grammatical articles or particles, all of which belong to the group of closed-class words. There aren't new function words created in the course of speech, so they can be defined as a fixed set of words even before having access to the data.

As there is not a defined standard of which are the official english function words, we have used the following list of 119 standard stop words for the experiment.

a	but	have	likely	own
able	by	he	may	rather
about	can	her	me	said
across	cannot	hers	might	say
after	could	him	most	says
all	dear	his	must	she
almost	did	how	my	should
also	do	however	neither	since
am	does	i	no	so
among	either	if	nor	some
an	else	in	not	than
and	ever	into	of	that
any	every	is	off	the
are	for	it	often	their
as	from	its	on	them
at	get	just	only	then
be	got	least	or	there
because	had	let	other	these
been	has	like	our	they

this	us	what	who	would
tis	wants	when	whom	yet
to	was	where	why	
too	we	which	will	you
twas	were	while	with	your

Note: Words like 'say', 'says', 'get', 'were', 'did', 'does', 'be', 'wants', etc. are included in this list as they are usually dropped in information retrieval or some natural language processing tasks because they are auxiliar verbs or they are words generally used to quote or to refer to something without having an apparent meaning in themselves, and thus not being part of the content words that are always desired for this tasks. Anyway, almost all of these words were not weighted between the most relevant words in the classifiers' output.

After we defined which function words to include we filtered out the IBM Communities dataset so all the words that were not present in this list were discarded. We call this the **FW dataset** (Function Words dataset).

B Automatic Classifiers

Among the classifiers provided by Weka, we decided to use seven different classifiers to cover the different most popular classification techniques: classifiers based on rules (ZeroR), based on decision trees (OneR, J48), bayesian classifiers (NaiveBayes), support vector machines (SMO) and regression models (Logistic Regression, Simple Logistic). We used the implementation provided by Weka (Witten & Frank, 2005) for each of the following algorithms:

Majority Class It is the simplest approach for the classification task, it consists of predicting all the forum posts as the role that has more presence in the dataset. In our case, 65.02% of the posts were written by members, so this algorithm predicts all the posts as member posts. Because is the simplest classification we can perform over the data we include this classification as a baseline to compare each of the other classifiers results, and measure their improvements as how far they results were from the majority class accuracy.

The reason why the majority class is different for some of the datasets shown in Table 2 it is because after applying the filters for each approach, there were some entries that became empty in their content, because all the words were discarded by the filter. In order to avoid the classifier to mislead the learning of each class by the absence of words as a characteristic of a given role to learn, we decided to discard these forum posts from the dataset reducing the total amount of posts, and thus changing the proportion of member-owner posts over the whole dataset.

OneR This is a simple classifier that builds a one-leaf decision tree with the word that considers to be the most predictive word for both roles, its output is a one level decision tree.

J48 This is another classifier based decision in decision trees that tries, through several levels, to make the tree fit the data as much as possible. Its output is a multi-level decision tree for each of the words in the vocabulary of the dataset that are considered relevant for the prediction.

Naive Bayes Naive Bayes is a classifier that uses estimator classes. It calculates numeric estimator values based on analysis of the training data. Its output consists of a list of all the words in the vocabulary and their means and standard deviation taken as probabilities for that word to appear on each class.

SMO SMO implements sequential minimal optimization algorithm for training a support vector classifier. The coefficients in the output are based on the normalized data, not the original data. Its output is a linear function of the regression model showing the attribute weights for all the words in the vocabulary.

Logistic Regression Logistic Regression is a classifier for building and using a multinomial logistic regression model with a ridge estimator. Its output consists of the linear function of the regression model showing the attribute weights for all the words in the vocabulary and a list of their odd ratios.

Simple Logistic Simple Logistic is a classifier for building linear logistic regression models. LogitBoost with simple regression functions as base learners is used for fitting the logistic models. The optimal number of LogitBoost iterations to perform is cross-validated, which leads to automatic attribute selection. Its output is made of the linear function of the regression model for both classes showing the attribute weights for the most relevant words in the model, each post is evaluated by both functions, the prediction is set by the function that gives the higher result.

C Detailed Results

C.1 Removing Outliers Approach Results

The Removing Outliers Approach was motivated by Gilbert's work (Gilbert, 2012) and it represents the basic information retrieval approach of removing the outliers tokens but from a simple 'bag of words' approach. The results obtained by this dataset are the nearest to the Human Performance only using unigrams, so it may be possible to get a better performance that the one attained by humans if we take into account ngrams, and features like the frequency of posts per author, or the size of the posts. In any case, being a domain dependent approach (because it takes into account only the words within the training dataset) this can be used as an upper bound baseline to test new approaches.

C.2 Enhancing through Stemming Approach Results

This approach was motivated by the basic technique in Information Retrieval and some NLP tasks that uses the stem of the words in order to be more efficient. In

fact, we propose it thinking that the words stems will increase the weight in stems of tokens that had the same meaning but different conjugation, like in the example given before that 'replies', 'reply' and 'replying' words will become one single token 'repli' with the addition of their weights as its own weight. This didn't seem to happen as we expected, decreasing in all the classifiers' accuracies respect to the Outliers Removed approach. It is possible that this happens because words endings are relevant for relations between the speakers, specially in languages like Spanish or French, so it makes sense that for this task in particular the accuracy is not increasing when using this approach. It is interesting to note that there was only one accuracy that increased significantly from the OR dataset results with this approach, and it was by the Logistic Regression classifier that increased from 61.82% in the OR dataset and below the Majority Class baseline (65.37%) to finish above that value (65.56%) for this S_OR dataset. Anyway, even though the best accuracy obtained for this dataset was 75.05%, making an improvement of 9.68% from the Majority Class baseline, the results were good enough to be used also as an upper bound baseline to compare new approaches.

C.3 Removing Outliers and Function Words Approach Results

This approach was motivated by another basic technique in Information Retrieval tasks that removes the function words from the dataset and focus only in the content words in order to achieve the task. In particular, we specially propose this 'only content words' approach because it represents the most domain dependent results we could get, and because we wanted to compare the FW dataset results with an approach that doesn't contain function words in its dataset. According to this basic approach, it is supposed to increase the accuracy respect to the OR dataset results, but we expected to decrease significantly showing evidence that the function words within the OR dataset were supporting the good results obtained by it. But in fact, none of this happened, even though the accuracy decreased from the OR dataset, it was not significantly different in at least the best two classifiers.

It is possible that this happens because domain dependent words that were not considered as relevant by the classifiers in the Outliers Removed approach were now weighted as relevant in order to maintain the accuracy as high as possible fitting to the dataset. Anyway, even though the best accuracy obtained for this dataset was 74.2%, making an improvement of 8.81% from the Majority Class baseline, the results were good enough to be used also as an upper bound baseline to compare new approaches.

C.4 Pronouns Approach Results

Since we started to explore the dataset we realized that there was a strong relation between the use of specific pronouns and each role. So this pointed out to pay attention to pronouns, but this approach was mainly motivated by one of the strategies of positive politeness which says that inclusive pronouns and 2nd person pronouns are used in a greater proportion when it is important to maintain a closer relationship with the hearer, and as this fits to the leadership relationship, we wanted to try out this approach in order to prove right or wrong this claim of politeness theory.

However, in the English language, pronouns are always included for almost all verb conjugations, becoming something that both roles use very often, so even when these words alone may not be good enough to predict the roles, two of the best results for this

dataset obtained an accuracy (66.92% and 66.9% respectively) significantly different from the Majority Class baseline as seen in the Table 4, showing enough evidence that leaders makes use of positive politeness strategies to communicate effectively.

C.5 Verbs vs Nouns Approach Results

Motivated by one of the claims of negative politeness which says that polite sentences have a higher proportion of nouns than verbs we wanted to try out this approach in order to prove if leaders make use of negative politeness strategies to communicate. Anyway, negative politeness is used normally when there is a social difference between the speaker and the hearer and depending in the leadership style this is not always the case. Implementing this approach, we didn't obtained useful results, we found that the proportion of nouns against verbs in member and owner posts didn't differ significantly, and even the best result obtained for this dataset by the Simple Logistic classifier (65.72%) wasn't significantly different from the Majority Class baseline. The results obtained mean that we don't have enough evidence to accept the claim that leaders make use of negative politeness strategies to communicate.

But this may also be caused by errors in the preparation of the experiment, in particular, it is important to note that the part of speech tagger is introducing noise during the tagging process where, specially in cases where it is difficult to determine the category of the word, for example: 'post' can be tagged as a verb and as a noun.

One example of this introduced error can be observed in the following tagged post from an owner forum post:

"('Hello', 'NNP') ('could', 'MD') ('anybody', 'VB') ('help','NN') ('me', 'PRP') ('please', 'VB') ('?', '.')"

Where NNP is a singular proper noun, MD is a modal verb, VB is a verb in the infinitive tense, NN is a single noun, and PRP is a personal pronoun.

As we can see in the example, 'Hello' is considered a proper noun maybe because of the capitalized first letter. But the great error here is when it's considering 'help' as a noun when it is a verb, and 'anybody' as a verb when it is a indefinite pronoun, and 'please' as a verb when in this sentence has the function of an adverb. Therefore, as it is clear, the problem of training automated classifiers over not previously annotated data is that we introduce a considerable amount of wrong tagged words that can mislead the final results.

C.6 Function Words Approach Results

This approach was motivated by Pennebaker's theory that function words are carriers of social meaning, specially because function words are the 'connective tissue' of the language so they are completely domain independent. Therefore, we wanted to prove this theory to complete both goals, prove this psycholinguistic theory and if possible present function words as the domain independent features we were seeking for the role inference task.

The best result obtained for this dataset by the Simple Logistic classifier achieved an accuracy of 69.12%, representing half of the way between the Majority Class baseline to the Human Performance accuracy. This is an interesting result, because it is common in Information Retrieval and some NLP tasks to discard function words as they are

considered irrelevant. With this results we have provided evidence that they alone are giving us considerable information to predict the roles, demonstrating that, in fact, function words are carriers of social meaning. There are also several advantages to take into account for function words: as a list of words of a fixed size (belonging to closed class of words) they are more efficient in terms of storage space and much more simple than calculating the filters in terms of computation complexity, that depending on the size of the dataset might consume considerable memory and time to apply. Also, as function words are domain independent, it allows us to have more reliability in the statistical model obtained from them, i.e no domain specific words, technical terms, or words related to particular communities were weighted as relevant in the output of the classifiers. And this property of being domain independent also allows, by the concept itself, to generalize into several different contexts.