Part of Speech Tagging with Naïve Bayes Methods

R. Creţulescu, A. David, D. Morariu, L. Vinţan
Computer Science and Electrical Engineering Department
"Lucian Blaga" University of Sibiu
Sibiu, Romania
{radu.kretzulescu, alexandru.david, daniel.morariu, lucian.vintan}@ulbsibiu.ro

***Abstract*— In this paper we have focused on the problem of automatic prediction of parts of speech in sentences. We present an experimental framework which includes the analysis and the implementation of methods for part of speech (POS) labeling (tagging). We have tested three methods that predict the POS without current word's context and also three context awareness statistic methods. The main goal of our work was to evaluate the three statistical methods Forward, Backward and Complete Method in order to analyze their applicability in the problem of automatically prediction of the POS. These methods are derived from the classic Naïve Bayes classifier. In our research we have used the WordNet database and a set of benchmarks called the Brown University Standard Corpus of Present - Day American English. The results obtained by the non-context-awareness methods compared to the results obtained by statistical methods are better but not so reliable like the statistical methods.**

***Keywords*— NLP, Naïve Bayes, Part Of Speech Prediction**

I. INTRODUCTION

In the field of Word Sense Disambiguation (WSD) [2] there were identified a range of linguistic phenomena such as preferential selection or domain information that are relevant in resolving the ambiguity of words. These properties are called linguistic knowledge sources. Current WSD system reports does not mention these sources but rather presents low-level features such as representations of "bag-of-words" or "n-grams" [3] used in disambiguation algorithms - one of the reasons being that the features (coding) incorporate more than one source of knowledge.

A lot of research in Natural Language Processing (NLP) [5] is focused especially on intermediate tasks that use known structures that are inherent in the language. Such a task is the part of speech labeling (or tagging). This process involves assigning a label to each word in a sentence; the label represents the part of speech (POS) of that word.

In this paper we have focused on the problem of automatic detection of the parts of speech within an English text in order to discover some semantic features of the sentence. For this, we have used the WordNet database [10] which is often used in WSD algorithms and a set of benchmarks called the Brown University Standard Corpus of Present - Day American English (or Brown Corpus) [1]. In this paper we present an experimental framework which includes the analysis and the implementation of algorithms for POS labeling. The POS prediction is a difficult task even for the English language because a significant part of English words (approximately 33%) are words that without a given context can have multiple syntactic forms (multiple parts of speech). Unfortunately these words are among the most commonly used in colloquial language. For example, tests on the Brown Corpus benchmarks [1] have obtained a frequency of 40% words [9] that have more than one POS. At the sentence/phrase level we have found that on average 60% of the words in a sentence have multiple parts of speech. This percentage is high enough to change the semantics of a sentence if the parts of speech are misunderstood. Solving this ambiguity is important because it is the basis for natural language processing applications such as machine translation and word disambiguation, etc. [7]. To this moment there is no automated solution for labeling parts of speech with perfect accuracy [8].

II. THE EXPERIMENTAL FRAMEWORK

A. The used datasets

We have developed our own framework which uses as input the Standard Corpus of Present-Day American English (Brown Corpus [1]). The Brown Corpus represents a general collection of texts that is used in natural language processing research and has been manually created by professors Henry Kucera and W. Nelson Francs.

We have divided the data provided by the Brown Corpus into two sets. A set containing 70% of texts, chosen randomly, will be used for training and the other set, which contains the remaining 30% of texts, will be used for testing. To simplify the usage and interpretation of the tags related to the words extracted from the Brown Corpus [1] we have decided to reduce the number of tags from the 82 tags to 5 tags. Our selected tags are: noun, verb, adverb, adjective and other (for any other POS). We chose those tags because we also used the WordNet database [10] which offer support only for the first four tags.

B. The application architecture

Our developed software architecture is specialized on the problem of grammatical analysis. It is created as a modular architecture which allows easy integration of disambiguation algorithms. The architecture provides a number of facilities for them such as extraction and pre-processing modules, a module for identifying tags for words using the WordNet and a module for evaluating the tagging accuracy.

The prediction of the POS is achieved using only WordNet or combination between WordNet and Brown Corpus. Also we

have implemented 3 context awareness statistical methods based on the Naïve Bayes classifier for predicting the part of speech.

C. Preprocessing

In order to be properly labeled the text submitted to the application must undergo several changes before it can be used for the tagging process. The first step is the fragmentation of the text documents in sentences (phrases). Each sentence is obtained by splitting the text after the punctuation mark (.,!, ?,) which is followed by a capitalized word. After identifying the sentences they pass through a preprocessing module which extends English grammar contractions (such as "I'm" becomes "I am", "it'll" becomes "It will" etc.). These abbreviations, if left as they are, will not be analyzed correctly by the classifier because those are different words merged into a structure.

D. Evaluation metrics

Measuring the performance of tagging methods is performed using and combining different evaluation metrics. These metrics indicates how good or correct the method is in order to predict a label for a particular word [4]. Our evaluation module integrates three different measures: precision, recall, f-measure [8]. Precision – represents the percentage of words correctly tagged into a category over those that have been tagged in that category. Recall - represents the percentage of relevant words correctly labeled into a category over all relevant words from that category. Precision and recall ranges from 1 to 0. In fact precision represents a quantitative measure of the information retrieval system while recall represents a qualitative measure of this system. F_measure – represents the harmonic mean between Precision and Recall [3, 6]. This measure penalizes better comparatively with other measures (as arithmetic mean) a system that is influencing more a metric over the other.

Also, in some cases, we use another metric for evaluating the algorithm called accuracy that we compute as number of correct tagged words divided by the total number of words that we had to label.

III. Methods for Tagging Words

A. Non Context-Awareness Tagging Methods

1) WordNet Prediction - WNP

The first proposed (simplistic, naïve) method uses only the WordNet database for predicting the POS. We considered these methods as being non context-awareness because they are not sensitive to the word's context (the word's neighbors). In this approach we use only the test set and compute the prediction accuracy for word tagging of the WordNet database considering each word in the sentence separately. If an independent word has multiple parts of speech in the WordNet we select only the POS that occurs most frequently in the WordNet. After all the extracted words have been tagged with a certain label (one out of four available in WordNet), these will be compared with existing labels in the Brown Corpus test dataset. A drawback of this dictionary noticed after checking its behavior used on the Brown Corpus benchmarks was its poor accuracy [9].

We have also checked all the options returned by WordNet with the correct solution (assuming that the Brown Corpus benchmark is perfect). If the correct solution is contained in the result provided by the WordNet dictionary we counted it as a correct prediction. Thus we were able to determine the maximum obtainable accuracy on the Brown Corpus set for a system which is determining the POS purely based on the WordNet. We have computed the accuracy as the ratio between the number of correct predictions over the total number of cases and we have obtained 72.93% [9].

2) Basic Global Probability - BGP

This simple method uses only the Brown Corpus database for predicting the POS. Thus using the training set extracted from the Brown Corpus we have counted separately for each word all parts of speech which appear for that word in the training data set. The BGP method computes the probability of a word to be a noun, a verb, an adverb, an adjective or another POS using the training set. These probabilities are obtained in the training phase and represent the ratio between the total number of occurrences of the given word, when it was labeled with a certain tag (noun, verb, adverb, etc.) and the count of all occurrences of the given word in the training set. We note this type of probability as P_{GP}. Thus for each word that appears in the training set we have compute five counters (for POS in which the word appears), so that at the end of training we can compute the most likely type of POS for that word.

Choosing the tag for a word in the test phase is made using the formula 3.1 (actually we are choosing the most probable type of POS for the word, computed in the training phase):

$$Label = \underset{pos \in \vartheta}{\operatorname{argmax}} (P_{GP_{pos}}) \qquad (3.1)$$

where $\vartheta=\{verb,noun,adv,adj,other\}$. The results of this method depend on how complex and distributed the training set is. Unfortunately this method will not be able to predict several types of speech for the same word. Also there is still a problem when a word appears in the test set and it was not present in the training set. In this case the word will be tagged with the label "not_found".

3) Sum Global Probability - SGP

The SGP method performs the prediction of the POS for the words extracted from the test set by adding the global probability obtained from the training set (probability computed in the previous section) and the probability obtained querying the WordNet database(P_{WN}). If the word in the test set has never occurred in the training set SGP uses only the probability obtained from the WordNet and if the WordNet doesn't provide any result (because it doesn't have the word in the database) the word will be labeled with "not_found".

$$Label = \underset{pos \in \vartheta}{\operatorname{argmax}} (P_{GP_{pos}} + P_{WN_{pos}}) \qquad (3.2)$$

4) Logarithm Global Probability - LGP

The LGP method predicts the parts of speech corresponding to the word in the test set by computing the product between the global probability BGP and the probability computed using WordNet. We observed that after combining the BGP with WordNet the results returned by BGP decrease. This can occur because the WordNet returns big values for probabilities and when summing these with small values returned by BGP the influence of WordNet is too strong. So we decided to try a method which multiplies these values. Because the computed probabilities can have in some cases very small values and multiplying them can raise the problem of small numbers we have used the monotone property of the logarithm: *log(x*y)=log(x)+log(y)*. So computing the logarithm of the probabilities and adding them, will not change the result.

$$Label = \underset{pos \in \vartheta}{\operatorname{argmax}} \left[\log_2(P_{GP_{pos}}) + \log_2(P_{WN_{pos}}) \right] \qquad (3.3)$$

All the methods presented up to this point don't adapt to the context in which the word occurs. The following statistical methods will be context-awareness.

B. Statistical Context-Awareness Tagging Methods

The methods presented in this section attempt to predict the POS for a word based on the context in which this word appears in the sentence. In order to include the context in which the particular word appears in the training set, we have computed some probabilities for training the Naïve Bayes classifier [6]. The Naïve Bayes classifier in this context does not compute the probability of appearance of a certain POS based on a certain class. Here it computes the probability of occurrence for the POS of a word conditioned by the POS of the previous word, next word or both words (word's locality).

In our approach we used the following general formula for the Naïve Bayes classifier.

$$pos_{NB} = \underset{1<i<5}{\operatorname{argmax}} \log \left[\begin{array}{l} P(Y_i) + \\ \sum_{j=1}^{5} \log P(x_j | Y_i) \end{array} \right] \qquad (3.4)$$

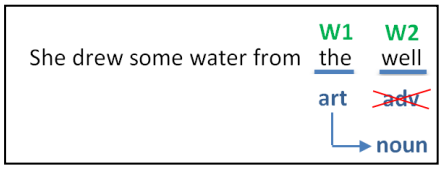
The index *i* represent the POS for the word that needs to be tagged. The index *j* represents the POS of the previous or of the next word (depending on the used method). The likelihood $P(Y_i)$ is a value that indicates the relative frequency for the label Y_i for the current word. The likelihood $P(x_j|Y_i)$ from the equation (3.4) computes how "suitable" the label Y_i (POS) is for the current word if the neighbor word has the tag x_j. The sum of these two probabilities is a value that indicates the POS for a particular word.

The equation computes the most significant label for a specific word. Because of the logarithm the obtained values are negative numbers.

1) Backward Naïve Bayes - BNB

This method computes the POS based on Global Probability taken from the training set (as in section above) and the POS from the previous word.

Let's take the following example:



In this example, if we take the word "well" and individually analyze it based only on the information retrieved from WordNet we will find that it is mostly used as an adverb, so for this word there is a small probability to choose as label another POS. From this point of view we have tried to influence the corresponding probability of a label associated to a particular word using the computed Naïve Bayes classifier probabilities.

After applying the BNB method the word "well" will be correctly labeled as a noun because of the article "the" to the left side of it.

To include in the prediction this aspect we have computed in the training step a matrix where we have stored the number of intersections of the two words *W1* and W2 which will be used by the Backward Naïve Bayes method in the testing step. Thus, at the end of the training step we have for each word a vector that contains the total number of occurrences of the particular word and its number of occurrences under different labels. In order to use also the WordNet we have extracted only 5 labels: noun (NN), verb (VB), adverb (RB), adjective (JJ) and other (OTH). The resulting matrix after training is a square matrix which has the dimension equal to the *no_of_words_in_the_training_set * no_of_the_parts_of_speech* and stores the number of intersections between any two words *W1* and *W2* for each label.

In the test phase we use the Backward Naïve Bayes method to predict the POS for the words extracted from the test set, based on the following information: the probability of the occurrence of certain parts of speech related to the current word and the conditional probability between the occurrence of words in some order and their appropriate types of the POS.

$$pos_{BNB} = \underset{i \in \vartheta}{\operatorname{argmax}} \left[\begin{array}{l} \log P(W2_i) + \\ \sum_{j \in \vartheta} \log P(W1_j | W2_i) \end{array} \right] \qquad (3.5)$$

where $\vartheta = \{VERB, NOUN, ADV, ADJ, OTH\}$

In the equation (3.6) where RB means adverb and VB means verb we exemplify how we compute the conditional probability from the equation 3.5. It expresses the probability that the word *W1* is an adverb conditioned that *W2* is a verb.

$$P(W1_{RB} | W2_{VB}) = \frac{\#\_(W1_{RB} \cap W2_{VB})}{\#\_(W2_{VB})} \qquad (3.6)$$

where $\#\_(W1_{RB} \cap W2_{VB})$ indicates how many times the two "words" $W1_{RB}$ and $W2_{VB}$ intersected (i.e. how many times the word *W1* appeared as adverb followed by the word *W2* as a verb) and $\#\_(W2_{VB})$ represents the number of occurrences of

the word *W2* as a verb.

For example, for two words *W1* and *W2* the intersection matrix would look like (Table I):

TABLE I. THE INTERSECTION MATRIX FOR BNB

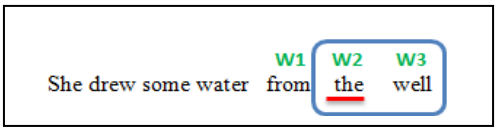
W1 \ W2	VB	NN	RB	JJ	OTH
VB	$W1_{VB} \cap W2_{VB}$	$W1_{VB} \cap W2_{NN}$	$W1_{VB} \cap W2_{RB}$	$W1_{VB} \cap W2_{JJ}$	$W1_{VB} \cap W2_{OTH}$
NN	$W1_{NN} \cap W2_{VB}$	$W1_{NN} \cap W2_{NN}$	$W1_{NN} \cap W2_{RB}$	$W1_{NN} \cap W2_{JJ}$	$W1_{NN} \cap W2_{OTH}$
RB	$W1_{RB} \cap W2_{VB}$	$W1_{RB} \cap W2_{NN}$	$W1_{RB} \cap W2_{RB}$	$W1_{RB} \cap W2_{JJ}$	$W1_{RB} \cap W2_{OTH}$
JJ	$W1_{JJ} \cap W2_{VB}$	$W1_{JJ} \cap W2_{NN}$	$W1_{JJ} \cap W2_{RB}$	$W1_{JJ} \cap W2_{JJ}$	$W1_{JJ} \cap W2_{OTH}$
OTH	$W1_{OTH} \cap W2_{VB}$	$W1_{OTH} \cap W2_{NN}$	$W1_{OTH} \cap W2_{RB}$	$W1_{OTH} \cap W2_{JJ}$	$W1_{OTH} \cap W2_{OTH}$

The disadvantage of this method is that it will predict the POS of a word based on the POS predicted for the previous word which can lead to an error propagation in case that at a moment it has predicted a wrong type of speech for a specific word. Nevertheless this method has the advantage that it is based on the type of speech of the previous word, fact that happens also in the natural language.

2) Forward Naïve Bayes - FNB

The FNB method will predict the POS of a word based on the global probability (BGP - the most POS for the given word) extracted from the training set and POS for the next word from the sentence.

For example, if we consider the sentence given below we try to predict the POS of the word *W2* based on the POS of the



word *W3* (the POS for the word *W3* must be also predicted).

To embed this prediction in the training phase we have used a matrix which stores the number of intersections between the two words *W2* and *W3* (example ($W3_{RB} \cap W2_{VB}$) which will be used by the Forward Naïve Bayes method. Thus, at the end of the training phase we have obtained for each word a vector which stores the total number of occurrences of that word and the occurrences for each POS for that word. Also a matrix is stored with the number of intersections between the two words *W3* and *W2* for each POS.

In the test phase we use the Forward Naïve Bayes method to predict the POS for the words extracted from the test set using the following formula:

$$pos_{FNB} = \underset{i \in \vartheta}{\operatorname{argmax}} \left[\begin{array}{l} \log P(W2_i) + \\ \sum_{j \in \vartheta} \log P(W3_j | W2_i) \end{array} \right] \quad (3.7)$$

where $\vartheta = \{VERB, NOUN, ADV, ADJ, OTH\}$

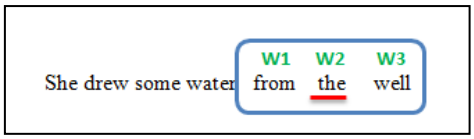
The computed probabilities are similar with those computed in the equation 3.5.

3) Complete Naïve Bayes - CNB

The Complete Naïve Bayes method will predict the POS for a word based on the global probability (the most significant POS for the current word), extracted using the training set and the probability for the appearance of a POS for the word which is positioned left and the word which is positioned right to the current word.

In the next example we will present the prediction of the POS (word *W2*) using the CNBs method taking into account the label of the previous and the next word.

To embed into the prediction process this aspect we used two matrices (like in Table I) in which we stored the number of intersections of the words *W1* and *W2* and the number of intersections between *W2* and *W3* for each POS.



Thus, at the end of the training phase we have for each word a vector containing the total number of occurrences of a particular word and the number of times it's labeled as verb, adverb, noun, etc. Also we compute two intersection matrices as described above.

In the test phase we use the Complete Naïve Bayes method to predict the POS for the words extracted from the test set using the following formula.

$$pos_{CNB} = \underset{i \in \vartheta}{\operatorname{argmax}} \left[\begin{array}{l} \log P(W2_i) + \\ \sum_{j \in \vartheta} \left[\begin{array}{l} \log P(W1_j | W2_i) + \\ \log P(W3_j | W2_i) \end{array} \right] \end{array} \right] \quad (3.8)$$

An advantage of this method is that it can compute the probability for the first word or for the last word of the sentence. The BNB has a problem because it can't predict the POS for the first word in a sentence and in this case it uses only the global probability for the first word. For this method it is very important to predict the POS for the first word correctly because all the next predictions in the sentence depend on it. The FNB method has same problem but with the last word from the sentence. This aspect is not so important here because a wrong prediction influences only the last word from the sentence.

Combining this two methods into the CNB for predicting the POS for the first word from the sentence we use only the FNB method and for the last word we use only the BNB method.

IV. EXPERIMENTAL RESULTS

The results were obtained based on the testing set extracted from the Brown Corpus and were evaluated using the metrics presented in Section II.D (precision, recall and F-measure). In the evaluation phase we considered that the Brown Corpus is perfectly labeled and we have referred to these labels. The

evaluation was made for each POS taken into consideration separately and we will also present an average value obtained by each method over all parts of speech.

A. WordNet Prediction

The next table (Table II) shows results only for the parts of speech supported by WordNet (noun, verb, adverb and adjective) using all the evaluation metrics. The discrepancy between the values of precision and recall suggests that the number of words classified as a certain POS is greater than the real number of words that belong to that POS. It also suggests that the total numbers of words that were classified correctly are very few compared with the total number of classified words.

TABLE II. RESULTS OBTAINING ONLY WORDNET PREDICTION (WNP)

WNP	Noun	Verb	Adverb	Adjective	Total
Precision	0.5822	0.7082	0.4365	0.4617	0.5522
Recall	0.7004	0.2346	0.6673	0.3281	0.4786
F-measure	0.6358	0.3525	0.5278	0.3836	0.5128

For this method we have computed also the maximum limit that could be obtained by this approach by taking in consideration all types of speech returned by WordNet and checking if one of them matches with the type written in the Brown Corpus test file. From the accuracy point of view the maximum obtainable value using this dataset was 72.93%.

B. Non Context-Awareness Methods Results

In this section we present results obtained by all non-context-awareness methods presented in Section III.A. The BGP (Basic Global Probability) method (Table III has achieved very good results and also very balanced metrics values.

TABLE III. RESULTS FOR BGP FOR EACH PART OF SPEECH

BGP	Noun	Verb	Adverb	Adjective	Other	Total
Precision	0.9564	0.9601	0.8977	0.9594	0.9793	0.9626
Recall	0.8548	0.9262	0.8194	0.9440	0.9822	0.9220
F-measure	0.9027	0.9428	0.8568	0.9516	0.9807	0.9419

The results obtained by the next two presented methods SGP (Sum Global Probability) and LGP (Logarithmic Global Probability) based on the global probability and the WordNet probability are lower. From the F-measure point of view the results are weaker with 39% for SGP and with 30% for LGP. This means that the WordNet diminishes the results obtained using only methods based on Brown Corpus. This occurs because the Brown Corpus returns values of probabilities that are very small comparatively with the probabilities returned by WordNet. In the Brown Corpus a word usually has all the parts of speech so the values of the probabilities are close to each other. In the WordNet a word usually has more appearances in a part of speech and fewer appearances in other parts of speech.

Also, the problem of WordNet is, that all the time, it returns the same values for the same word (the most frequently POS for given word). The Brown Corpus dataset contains common words that typically are not in the part of speech usually returned by WordNet that is why the methods based on WordNet obtained so weak results.

TABLE IV. RESULTS OBTAINED BY SGP FOR EACH POS SEPARATELY

SGP.	Noun	Verb	Adverb	Adjective	Other	Total
Precision	0.699	0.902	0.497	0.730	0.969	0.742
Recall	0.733	0.479	0.692	0.311	0.189	0.439
F-measure	0.716	0.626	0.579	0.437	0.317	0.552

TABLE V. RESULTS OBTAINED BY LGP FOR EACH POS SEPARATELY

LGP.	Noun	Verb	Adverb	Adjective	Other	Total
Precision	0.8171	0.9216	0.8522	0.8899	0.8896	0.8620
Recall	0.7232	0.4598	0.6875	0.3851	0.3926	0.5098
F-measure	0.7673	0.6135	0.7611	0.5376	0.5448	0.6407

Even if the BGP method obtains very high and balanced results it does not mean that this is the best method for predicting the POS. It demonstrates us that the Brown Corpus benchmark contains mainly words that belong to a single POS regardless of the context. In the BGP method we compute, based on the training set, the POS probability for each word and return always in the testing phase the same POS for a given word (POS with the highest probability all the time). From the experimental results we have observed that the POS returned by this method differ from the POS returned by WordNet. We can conclude that in the Brown Corpus the words are used frequently with the same POS and so that for other texts we expect that this method will not return good results.

C. Context-Awareness Statistical Methods Results

In this section we present the results obtained by the statistical methods for computing the POS. The statistic, computed as in the Naïve Bayes classifier, is computed based only on the training set. In the training part we count all pairs of two and three words with the corresponding POS. We present results for all the three methods described in Section III.B. As we expected (because a POS of a word depends in almost all the cases on the POS of the previous word) among the three implementations the Backward version obtains the better results (Table VI). From F-measure point of view with this method we obtain 73.3% in average.

TABLE VI. RESULTS OBTAINED BY BACKWARD NAÏVE BAYES

BNB	Noun	Verb	Adverb	Adjective	Other	Total
Precision	0.7	0.7217	0.4099	0.8662	0.8031	0.7418
Recall	0.7175	0.7209	0.5258	0.6912	0.7803	0.7244
F-measure	0.7086	0.7213	0.4607	0.7688	0.7915	0.733

The results obtained using the methods Forward and Complete Naïve Bayes showed a reduction of 2% respectively 4% compared to Backward version. However the metrics values are more balanced values compared to those obtained by SPG and LPG methods.

TABLE VII. RESULTS OBTAINED BY FNB FOR ALL POS

FNB	Noun	Verb	Adverb	Adjective	Other	Total
Precision	0.6514	0.7317	0.4525	0.8605	0.7785	0.719
Recall	0.7464	0.7196	0.5868	0.6776	0.6839	0.7021
F-measure	0.6957	0.7256	0.5110	0.7582	0.7281	0.7105

TABLE VIII. RESULTS OBTAINED BY CNB FOR EACH POS SEPARATELY

CNB	Noun	Verb	Adverb	Adjective	Other	Total
Precision	0.6471	0.7284	0.3561	0.8601	0.7729	0.7060
Recall	0.7415	0.7126	0.5215	0.6856	0.6576	0.6894
F-measure	0.6911	0.7204	0.4232	0.7630	0.7106	0.6976

Even if the results are not as good as those obtained by the BGP method (but better than SGP and LGP) we can say that the Naïve Bayes method in all three implemented versions is more stable and less depending on word context comparatively with the Global probability method.

We think that the poor results obtained by Naïve Bayes methods have the causes in the prediction part. If we incorrectly predict the POS for the current word, the predictions of the POS of the next words have high probability to be incorrectly predicted, too.

For these three statistical methods we have performed also a separate experiment to compute the maximal limit that can be obtained if in all cases when we predict the POS for the current word, the POS for the neighboring words is correct. In this experiment, in the testing phase, after we predict the POS for the current word using one of the three methods, we test if the prediction is correct. If the prediction was ok we marked the success and made no modification. If the prediction was wrong, we marked the failure and modify the POS for the current word with the correct value based on the Brown Corpus and went further. This modification implies that the error of the predicted POS will not propagate further in the next prediction. In the Table IX we present the results obtained with this "ideal" approach for all methods based on Naïve Bayes implementation and consider those results as the "maximum limits" that can be obtained using these methods.

TABLE IX. THE MAXIMUM LIMIT FOR NAÏVE BAYES METHODS

		Noun	Verb	Adverb	Adj.	Other	Total
BNB	Precision	0.95	0.94	0.89	0.96	0.98	0.96
	Recall	0.89	0.93	0.83	0.96	0.98	0.93
	F-measure	0.92	0.94	0.86	0.96	0.98	0.95
FNB	Precision	0.95	0.95	0.89	0.96	0.98	0.96
	Recall	0.89	0.93	0.85	0.96	0.98	0.94
	F-measure	0.92	0.94	0.87	0.96	0.98	0.95
CNB	Precision	0.94	0.93	0.86	0.96	0.98	0.95
	Recall	0.88	0.90	0.83	0.95	0.98	0.93
	F-measure	0.91	0.91	0.85	0.95	0.98	0.94

V. CONCLUSIONS

In this paper we have evaluated two types of methods in order to analyze their applicability in the problem of automatic POS prediction. The method based only on WordNet achieved an overall prediction probability of only 51.28%. The POS with the weakest prediction is the verb obtaining a value of only 35.25%.

The Basic Global Probability method has obtained an overall prediction probability of 94%; the results are very good and also the results for the evaluation metrics are balanced. However, these values do not suggest that this is the best method for the POS prediction but demonstrates that this method is highly dependent on the set used for training. Because the BGP methods obtains better results than the SGP and LGP methods we can conclude that the Brown Corpus benchmark contains mainly words that belong to a single part of speech regardless of the context.

The Backward Naïve Bayes method obtains an average value of 73% (according to F-measure). The Forward and Complete Naïve Bayes versions have a prediction rate of only 71% and 69% (according to F-measure). However the results obtained by all metrics are more balanced values compared to those obtained by SGP and LGP.

The balanced values provided by Naïve Bayes methods suggest that all three implementation variants are more stable and less dependent on the context used in the training phase comparatively with the global probability method.

As further work we propose to use these methods in the document representation phase for text classification. Until this moment we represent text document as bag of words without any syntactic information (we represent documents only as word frequency vectors).

Acknowledgement

This work was partially supported by the strategic grant POSDRU/159/1.5/S/133255, Project ID 133255 (2014), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007-2013.

VI. REFERENCES

[1] Brown University Standard Corpus of Present-Day American English (Brown Corpus) http://icame.uib.no/brown/bcm.html, accessed in June 2013;

[2] E. Agirre, P. Edmonds, Word Sense Disambiguation. Algorithms and Applications, Springer, 2007, ISBN 978-1-4020-4808-4;

[3] J. Han, M. Kamber, J. Pei, Data Mining: Concepts and Techniques, Third Edition, Morgan Kaufmann Publishers, 2012, ISBN 9780123814791;

[4] D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008, ISBN-10: 0521865719;

[5] D. Manning, H. Schutze, Foundations of Statistical Natural Language Processing, MIT Press, ISBN: 987-0-262-133360-9, 1999;

[6] M. Mitchell, Machine Learning, McGraw-Hill Science/Engineering/Math, 1997, ISBN: 0070428077;

[7] R. Mitkov, The Oxford Handbook of Computational Linguistics, Oxford University Press, ISBN: 0-19-823882-7, 2003;

[8] D. Morariu, R. Cretulescu , L. Vintan, Vector versus Tree Model Representation in Document Clustering, Romanian Journal of Information Science and Technology (ROMJIST), vol. 16, no. 1, pp. 81-102, ISSN: 1453-8245, Romanian Academy, Bucharest, 2013;

[9] D. Morariu, R. Cretulescu, M. Breazu, Word Sense Disambiguation for Text Mining, The third international conference in Romania of "Information Science and Information Literacy", ISSN 2067-9882, April 2012;

[10] Princeton University WordNet – A lexical database for English http://wordnet.princeton.edu, accessed in June 2013;