At-least-N voting over biomedical named entity recognition systems

Manabu Torii[1] and Hongfang Liu[2]
[1]ISIS Center, University Medical Center, Washington, DC 20057 USA.
[2]Department of Biostatistics, Bioinformatics and Biomathematics, and the Protein Information Resource, Georgetown University Medical Center, Washington, DC 20057 USA.

ABSTRACT

Biomedical named entity recognition (BNER) has been actively studied over the years, and several BNER systems have become publicly available. In this study, we investigate the utility of a simple voting method called at-least-n voting to improve gene name recognition, which takes advantage of the availability of BNER systems in the domain. We found this voting scheme is effective in combining BNER systems, and furthermore a combined system derived with publicly available BNER resources can be competitive with that of state-of-the-art gene recognition systems. The study implies that system combination utilizing diverse techniques and resources is very promising for BNER.

1 INTRODUCTION

Over the years, named entity recognition (NER) has been studied actively in the biomedical language processing field, and several biomedical NER (BNER) systems have become publicly available, e.g., (Fukuda et al. 1998; Tanabe and Wilbur 2002; Settles 2004; Leaman and Gonzalez 2008). These systems exploit domain-oriented features such as distinctive affixes of biological entity names, e.g., "-amide" and "-cyte", and in-domain lexical resources such as protein names in sequence databases. In the future, there may be more choices of BNER systems as new techniques and resources are introduced to the field. Given diverse BNER systems in the field, system combination is of great interest to boost BNER performance.

System combination has been a viable solution to enhancing the performance of classification systems (Dietterich 2000; van Halteren 2001) including NER systems (Florian et al. 2003). In the biomedical domain, (Si et al. 2005)combined systems that participated in the JNLPBA shared task, recognition of five types of entities in MEDLINE abstracts, and reported excellent performance using Conditional Random Fields (CRFs). (Wilbur et al. 2007) combined 21 systems from the BioCreAtIvE II Gene Mention (GM) task, and reported an F-measure over 90% using CRFs. Unlike these approaches that use another level of machine learning, in this study, we investigate a simple voting scheme that could be readily implemented on top of available NER/BNER systems without the need of training. This particular voting method was named *at-least-n voting* in (Kambhatla 2006), and was used with bagging technology (Breiman 1996) over homogeneous systems. We experimented at-least-n voting for varying *n* over heterogeneous systems that participated in the BioCreAtIvE II GM task. We further examined if the method can be applicable to systems built on publicly available NER/BNER resources.

In the following, we use the term gene to refer both gene and protein that share the same surface strings.

2 BACKGROUND

2.1 Biomedical Named Entity Recognition

Early BNER systems were hand-crafted rule/pattern-based systems that encode expert knowledge of biomedical entity names, e.g., (Fukuda et al. 1998). As large annotated corpora became available in the domain, machine learning frameworks have been introduced to BNER, and they have demonstrated excellent performance in reproducing human annotation of gene names in text (Kim et al. 2004; Yeh 2005; Wilbur et al. 2007). Among other machine learning frameworks, CRFs have become a popular solution to BNER for its excellent recognition performance and also for the availability of its implementation, e.g., MALLET (McCallum 2002). ABNER, based on MALLET, is one of the early applications of CRFs to BNER. It exploits domain-oriented features such as semantic type features with a first order CRF model. The source code of ABNER is available online (http://pages.cs.wisc.edu/~bsettles/). Recently, another MALLET-based system BANNER became available (http://banner.sourceforge.net/), which exploits a large number of features in a second order CRF model. LingPipe suite by Alias-*i*, among its versatile language processing capabilities, allows users to build different types of NER systems. Without tuning to biomedical text, a system derived with LingPipe still performs well for gene name recognition (Carpenter 2007).

The choice of machine learning algorithm affects performance of the resulting recognition systems, but the selection of features and the use of domain dictionaries is also an important aspect in improving recognition performance. A number of systems in the BioCreAtIvE II GM task incorporated lexical entries from online resources, e.g., Locus Link, HUGO, UniProt/SwissProt. Dictionary lookup can also build BNER systems, e.g., (Tsuruoka 2004; Liu et al. 2006).

System combination is an effective solution to boost BNER performance. In the BioCreAtIvE GM task, some systems combined models trained for forward and backward parsing directions, models trained on different annotation boundaries, or trained with different learning algorithms. In most of these systems, combinations were based on set-union of names recognized by different models (Wilbur et al. 2007).

2.2 At-least-n voting

In at-least-n voting for system combination (Kambhatla 2006), one target class is assumed among available

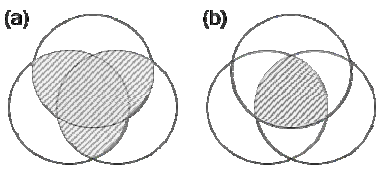

Figure 1. Graphical representation of at-least-n voting. (a) m=3 and n=2 and (b) m=3 and n=3. Each oval represents a set of instances, and the shaded areas represent the sets resulted from voting.

classes. Each participating system gets one vote, and a minimum of n votes will qualify an instance to be categorized as the target class. There are two variables for at-least-n voting: the number of systems participating in voting, m, and the minimum votes, n. Figure 1 shows the graphical representation of the method for (m, n) = (3, 2) and (3, 3).

In some classification tasks, the number of positive instances is much smaller than that of negative instances. In such situations, at-least-n voting may be preferred to the widely used majority voting strategy, in which a class with the majority votes is assigned to an instance. Kambhatla tested both at-least-n voting and majority voting in a bagging setting to improve classification of relation between two named entities for Arabic and Chinese text. Specifically, m maximum entropy classifiers were trained on sampled training instances, and outputs of the classifiers were combined through voting. In the experiments, at-least-n voting performed better than majority voting.

2.3 BioCreAtIvE Gene Mention Corpus

In this study, we used the BioCreAtIvE Gene Mention (GM) corpus introduced in the shared-task challenges BioCreAtIvE I and II. The entire corpus consists of 20,000 sentences from diverse MEDLINE abstracts, in which gene names are manually annotated. During BioCreAtIvE II challenge, 15,000 sentences were provided as training data for the GM task, and the remaining 5,000 sentences were used for evaluation. A participating team in the GM task submitted at most three runs (i.e., three sets of gene names detected in the test corpus). The performance of each system (i.e., a submitted set) was measured by precision, recall, and F-measure. After the workshop, the entire corpus and the sets of gene names submitted by the participants were publicly released (http://biocreative.sourceforge.net/).

For each participating team, we selected one set of gene names that marked the highest F-measure among their submitted sets, and thus we selected 21 sets for our study. In the following, we denote these sets as S_i for i=1, 2, …, 21, where the subscript i is the F-measure rank of the set. S_1 is the set yielding the highest F-measure among the 21 sets, which is 87.2% (precision/recall of 88.5/86.0%). Details of these systems can be found in (Wilbur et al. 2007).

3 EXPERIMENTS AND RESULTS

In the following experiments, we applied at-least-n voting for groups of m sets selected from the 21 sets resulted in the BioCreAtIvE II challenge. We assume recognized names in these sets as distinctive instances in the voting mechanism, and thus each system votes for a phrase (a word or a sequence of words), and not for an individual word to label if it is a part of a gene name or not.

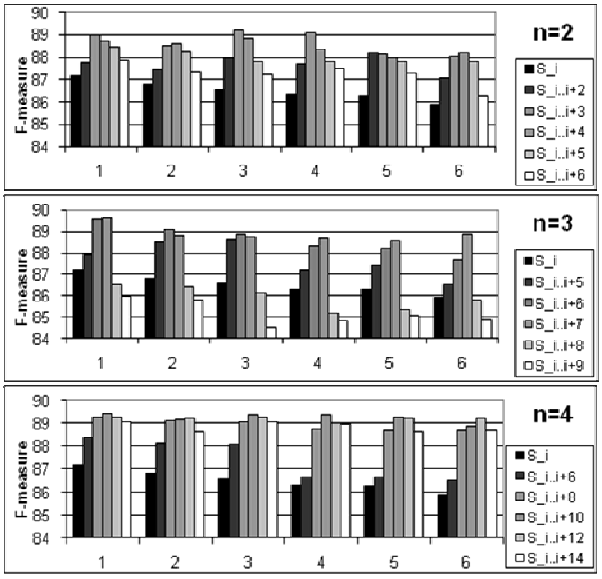

Figure 2. Performance of voted systems. X-axis is i, e.g., the leftmost bar for a particular i (the darkest bar) shows the F-measure of S_i, followed by the bar shows the F-measure of a voted system, where S_i, S_{i+1}, …, S_{i+m-1} are combined through at-least-n voting.

3.1 General effects of at-least-n voting

We are interested if at-least-n voting can generally improve the performance of BNER systems, and how to select variables (m, n) or systems participating in voting. To investigate this aspect, we applied the method to S_i, S_{i+1}, …, S_{i+m-1} for $0 \le i \le 21$, while varying m and n. We applied the method to consecutive sets in the F-measure rank order because intuitively it was discouraged to combine sets with very different F-measures (This is revisited in the next experiment). Figure 2 shows the results for n=3, 4 and 5 for varying m. The highest F-measure observed was 89.6% (precision/recall of 88.6/90.7%) for (m, n)=(7, 3) and the participating systems were S_1 to S_6, which is 2.4% higher than the F-measure of S_1.

Next, for fixed (m, n), we applied the method for all possible groups of sets, and observed the relation between the difference of the F-measures (the largest difference among the sets in the group), i.e., $F_b - F_w$, where F_b and F_w are the best and worst F-measure among the constituent sets, respectively, and the percent improvement (or degradation) of the combined system, i.e., $100 \times (F_c - F_b)/F_b$, where F_c is the F-measure of the combined set. Figure 3 shows the plot of the paired values when (m, n)=(4, 2), thus, for C(21, 4)=5,985 groups of four sets. The highest F-score observed was 90.1% (precision 90.7% and recall

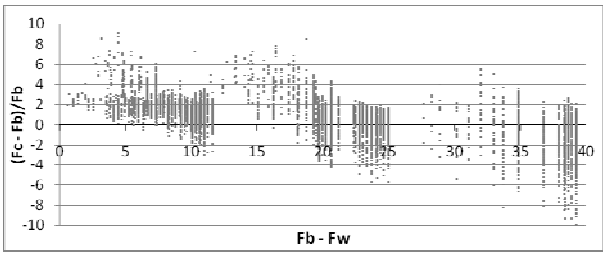

Figure 3. Relation between the F-measure difference among the participating systems (x) and the improvement in the F-measure (y).

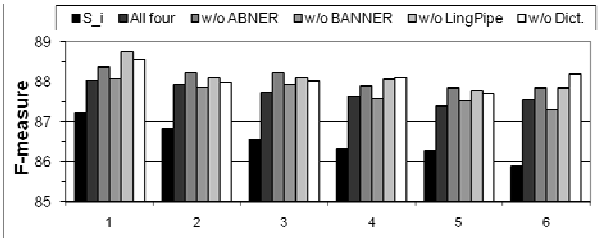

Figure 4. Boosting a customized BNER system with publicly available BNER/NER sources using at-least-2 voting.

89.4%) when combining S_1, S_2, S_6, and S_{17}, which is 2.9% higher than the F-measure of S_1. Table 1 shows the top 5 F-measures observed in this process and the corresponding four sets constituting the combined sets.

3.2 At-least-n voting over systems derived with publicly available BNER resources

We are also interested if the method is applicable to systems trained with publicly available NER/BNER packages. We used BANNER (ver. 0.2) with a pre-compiled model trained on the BioCreAtIvE training corpus. We also trained systems using ABNER package (ver. 1.5) and LingPipe suite (ver. 3.1.2) over the training portion of the GM corpus. As for LingPipe, we trained a CharLmRescoringChunker model by following an example in the online tutorial. We only changed the size of the n-gram from 12 to 36. For tokenization, we used a program in ABNER. Besides these machine learning systems, we used dictionary lookup to find gene names in the test corpus. As a gene name dictionary, BioThesaurus (ver. 4.0), a thesaurus of gene/protein names derived from 35 online resources, was used (Liu et al. 2006). By tokenizing and normalizing both the dictionary and the input text, we emulated flexible lookup (Tsuruoka 2004). To mitigate false positive problems typical of dictionary lookup, prevalent false positive phrases were identified in the training corpus (i.e., find a dictionary entry whose occurrences in text are not annotated as gene for > 5%), and they were removed from the dictionary.

Table 2 shows the performance of the individual systems on the test corpus[1], and that for the voted systems with n=2 and m=3 and 4. We only tested the case for n=2 because we should use more than four systems in the voting committee to expect improvement for $n \geq 3$. Combination of all four systems achieved an F-measure of 87.6% (precision/recall of 86.1/89.1%), which is competitive with the F-measure of S_1 (87.2%).

We further investigate if these systems could be used in boosting a customized high performance system, i.e., a particular set from the 21 sets. Motivation behind the experiment is to test if publicly available resources are still useful in boosting the recognition performance when one can afford to develop an in-house high-performance system. Each of the 21 sets was combined with these four systems, and also with every three of the four systems using at-least-2 voting. The results for S_1 to S_6 can be found in Figure 4. The best results observed was the F-measure of 88.7% (precision/recall of 88.0/89.5%) when combining S_1 with the three other systems derived with ABNER, BANNER, and BioThesaurus.

Table 1. Top 5 F-measures and their corresponding groups.

	Precision	Recall	F-measures	Sets
1	90.7	89.4	90.1	S_1, S_2, S_6, S_{17}
2	89.5	90.5	90.0	S_1, S_3, S_6, S_{17}
3	89.0	90.8	89.9	S_1, S_2, S_4, S_6
4	89.8	89.9	89.9	S_1, S_3, S_6, S_7
5	90.9	88.8	89.8	S_1, S_2, S_6, S_7

Table 2. Performance of publicly available NER/BNER resources, and their combination through at-least-2 voting.

	Precision	Recall	F-measures
1. ABNER	85.9	78.4	82.0
2. BANNER	87.4	82.8	85.0
3. LingPipe	79.0	86.1	82.4
4. BioThesaurus	55.1	85.5	67.0
1, 2, and 3	90.3	80.9	85.3
1, 2, and 4	91.0	80.6	85.5
1, 3, and 4	88.5	80.5	84.3
2, 3, and 4	88.9	83.1	85.9
1, 2, 3, and 4	86.1	89.1	87.6

4 DISCUSSION

Comparing to system combination based on machine learning as in (Wilbur et al. 2007), at-least-n voting is much easier to deploy without the need of training and achieves comparable performance. Our study implies that given a group of systems where each has relatively good performance, the combined system using at-least-n voting generally outperforms each of the participating systems. As shown in Figure 3, the voting method is effective in improving the performance of gene recognition systems especially when the F-measure differences among the participating systems are less than 5% (Figure 3).

We also observed that the method was applicable to BNER systems derived with publicly available tools/resources, and such systems could be used to boost the performance of customized high-performance systems as well. In certain applications, we should pay attention to the trade-off between the precision and the

[1] The F-measures of the ABNER and LingPipe models are higher than those reported on the same data set by (Baumgartner et al. 2007), who also included these models in simple voting schemes. They used pre-trained models included in the packages, and did not build new models over the GM training corpus.

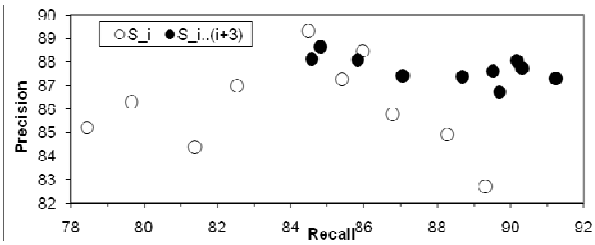

Figure 5. Precisions and recalls for S_1… S_{10} and for voted systems combining S_i, …, S_{i+3} with (m, n)=(4, 2).

recall. For example, at-least-2 voting generally has an advantage in boosting recalls (Figure 5), while we can boost precisions by increasing n.

Notable observation was systems similar to each other have limited contribution when they are combined. For example, two sets S_2 and S_3 have a relatively large overlap. For these sets, the proposed combination method requires either one of the two sets in achieving a high F-measure as in Table 1. Similarly, as for S_6 that were derived with a second order CRF model incorporating BioThesaurus as a domain lexicon, it is discouraged to combine it with BioThesaurus lookup results again (Figure 4). On the contrary, the voting scheme was effective when the constituent systems were diverse. For example, the four sets S_1, S_2, S_6, and S_{17} made a voted system with the F-measure 90.1%, where they were regularized linear classifiers exploiting unlabeled MEDLINE abstracts (S_1), CRF models with a number of lexical features (S_2), a CRF model with BioThesaurus supplemented with LingPipe outputs (S_6), and a rule-based system encoding domain lexical knowledge (S_{17}). In the same experiment, among the top 50 (100) of 5,985 groups of fours, S_1 appears 35 (62) times, S_2 appears 20 (31) times (S_3 appears 21 (44) times), S_6 appears 42 (79) times, and S_{17} appears 21 (31) times. Frequent appearance of S_6 in the top 50 (100) list may be credited to BioThesaurus, which makes the large contribution to the performance of S_6 (a manuscript in preparation). Also, frequent appearance of S_1 may imply its unique contribution through the exploitation of unlabeled MEDLINE abstracts.

5 CONCLUSION

Classification combination is a viable solution to enhancing the performance of BNER systems. In this study, we showed the utility of simple voting called at-least-n voting in improving the performance of gene name recognition systems.

We found an important consideration in applying at-least-n voting is the selection of participating systems. We observed that machine learning models derived with the same learning algorithm (e.g., CRFs used in ABNER, BANNER, S_2, S_3, S_4, or S_6) could still be useful during at-least-n voting, which conforms to the results by (Kambhatla 2006). Meanwhile, we found the voting scheme can be very effective when constituent systems are functionally diverse. Particularly, we identified three NER approaches to achieving system diversity: the use of unlabeled text (e.g., S_1), dictionary resources (e.g., S_6), and rules/patterns encoding expert knowledge (e.g., S_{17}). In the future study, we will look for BNER systems exploiting these aspects to improve the performance of voted systems, while we seek NER approaches orthogonal to these three approaches.

ACKNOWLEDGEMENTS

We greatly appreciate those who made the corpora and the NLP tools publicly available for the research community. This project was supported by IIS-0639062 from NSF.

REFERENCES

Breiman, L. (1996) Bagging predictors. *Machine Learning*, 24, 123–140.

Baumgartner, W.A. Jr., Z. Lu, H.L. Johnson, J.G. Caporaso, J. Paquette, A. Lindemann, E.K. White, O. Medvedeva, K.B. Cohen, and L. Hunter (2007) An integrated approach to concept recognition in biomedical text, *In Proc of the Second BioCreative Challenge Evaluation Workshop*, pp. 307–309.

Carpenter, B. (2007) LingPipe for 99.99% Recall of Gene Mentions, *In Proc of the Second BioCreative Challenge Evaluation Workshop*, pp. 307–309.

Dietterich, T.G. (2000) Ensemble Methods in Machine Learning, *In Proc of the First International Workshop on Multiple Classifier Systems*, pp. 1-15.

Florian, R., A. Ittycheriah, H. Jing and T. Zhang (2003) Named entity recognition through classifier combination, *In Proc of Natural language learning at HLT-NAACL*, pp.168–171.

Fukuda, K., T. Tsunoda, A. Tamura and T. Takagi (1998) Toward information extraction: identifying protein names from biological papers. *In Proc of PSB*, pp. 705–716.

Kambhatla, N. (2006) Minority vote: at-least-N voting improves recall for extracting relations, *COLING/ACL poster*, pp. 460 – 466.

Kim, J.D., T. Ohta, Y. Tsuruoka, Y. Tateisi and N. Collier (2004) Introduction to the bio-entity recognition task at JNLPBA. *In Proc of the International Workshop on Natural Language Processing in Biomedicine and its Application*.

Leaman, R. and G. Gonzalez (2008) BANNER: An executable survey of advances in biomedical named entity recognition, *In Proc of PSB*, 13:652-663

Liu, H., Z.Z. Hu, M. Torii, C. Wu, C. Friedman (2006) Quantitative assessment of dictionary-based protein named entity tagging, *J. Am. Med. Inform. Assoc.*, 13(5):497–507.

McCallum, A.K. (2002) MALLET: A Machine Learning for Language Toolkit, http://mallet.cs.umass.edu.

McDonald, R.T., R.S. Winters, M. Mandel, Y. Jin, P.S. White and F. Pereira (2004) An entity tagger for recognizing acquired genomic variations in cancer literature, *Bioinformatics*, 20, 3249–3251.

Settles, B. (2004) Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets. *In Proc of COLING 2004*, pp. 104–107.

Si, L., T. Kanungo and X. Huang (2005) Boosting Performance of Bio-Entity Recognition by Combining Results from Multiple Systems, *In Proc of BIOKDD*, pp. 76–83.

Tanabe L. and W.J. Wilbur (2002). Tagging gene and protein names in biomedical text. *Bioinformatics*, 18(8):1124–1132.

Tsuruoka, Y. and T. Jun'ichi (2004) Improving the Performance of Dictionary-based Approaches in Protein Name Recognition, *J Biomed Inform*, vol 37, Issue 6, pp. 461-470.

van Halteren, H., J. Zavrel and W. Daelemans (2001) Improving accuracy in word class tagging through the combination of machine learning systems. *Computational Linguistics*, 27(2):199–230.

Wilbur, J., L. Smith and L. Tanabe (2007) BioCreative 2. Gene Mention Task, *In Proc of the Second BioCreative Challenge Evaluation Workshop*, pp. 7–16.

Yeh, A., S. Alexander, A. Morgan, M.E. Colosimo and L. Hirschman (2005) BioCreAtIvE Task 1A: Gene Mention Finding Evaluation, *BMC Bioinformatics*, 6 Suppl.