The Tanl tagger for Named Entity Recognition on Transcribed Broadcast News

Giuseppe Attardi, Giacomo Berardi, Stefano Dei Rossi, Maria Simi

Università di Pisa, Dipartimento di Informatica, Largo B. Pontecovo 3,
56127 Pisa, Italy
{attardi, berardi, deirossi, simi}@di.unipi.it

Abstract. The Tanl tagger is a configurable tagger based on a Maximum Entropy classifier, which uses dynamic programming to select the best sequences of tags. We applied it to the NER tagging task, customizing the set of features to use, and including features deriving from dictionaries extracted from the training corpus. The final accuracy of the tagger is further improved by applying simple heuristic rules.

Keywords: Named Entity Recognition, Maximum Entropy, dynamic programming, chunker

1 Description of the System

The Tanl tagger is a generic, customizable text chunker, which can be applied to tasks such as POS tagging, Super-sense tagging and Named Entity recognition [1]. The tagger, based on the work of Chieu & Ng [2], uses a Maximum Entropy classifier for learning how to chunk texts. Maximum Entropy is a more efficient technique than Support Vector Machines (SVM): by complementing it with dynamic programming it can achieve similar levels of accuracy.

The tagger has an option (called *refine*) to transform the IOB annotations into a more refined set of tags: the B tag is replaced by U for entities consisting of a single token; the last I tag of an entity of more than one token is replaced by E. Experiments have shown that for NER the refinement is effective, helping the classifier to better separate the data.

Since the Maximum Entropy classifier assigns tags to each token independently, it may produce inadmissible sequences of tags. Hence a dynamic programming technique is applied to select correct sequences. A probability is assigned to a sequence of tags t_1, t_2,…, t_n for sentence s, based on the probability of the transition between two consecutive tags $P(t_{i+1} \mid t_i)$, and the probability of a tag $P(t_i \mid s)$, obtained from the probability distribution computed by Maximum Entropy:

$$P(t_1, t_2, ..., t_n) = \prod_{i=1}^{n} P(t_i \mid s) P(t_i \mid t_{i-1})$$

In principle the algorithm should compute the sequence with maximum probability. We use instead a dynamic programming solution which operates on a window of size $w = 5$, long enough for most super-senses. For each position n, we compute the best probability $PB(t_n)$ considering the n-grams of length $k < w$ preceding t_n:

$$PB(t_n) = \max_k PB(t_{n-k-1}) \ldots PB(t_{n-1})$$

A baseline is computed, assuming that the k-gram is made all of ‘O’ (outside) tags:

$$PB_O(t_n) = \max_k PB(t_{n-k-1})\ P(t_{n-k} = \mathrm{O}) \ldots P(t_{n-1} = \mathrm{O})$$

Similarly for each class C we compute:

$$PB_C(t_n) = \max_k PB(t_{n-k-1})\ P(t_{n-k} = \mathrm{C}) \ldots P(t_{n-1} = \mathrm{C})$$

and finally:

$$PB(t_n) = \max(PB_O(t_n), \max_C PB_C(t_n))$$

1.1 Features Specification

The modular architecture of the chunker relies on a textual configuration file. In particular three different kinds of features can be extracted:

- **attributes features:** represent certain attributes (e.g.: PoS, Lemma, NE) of surrounding tokens, expressed by the relative positions w.r.t. to the current token; for example POSTAG -1 0 means: use as context features for the current token the PoS of the previous token and of the current token, in position 0;
- **local features:** other binary morphological features extracted from the analysis of the current word and the context in which it appears; for example “*previous word is capitalized*”;
- **global features:** properties holding at the document level. For instance, if a word in a document was previously annotated with a certain tag, then it is likely that other occurrences of the same word should be tagged similarly. Global features are particularly useful in cases where the word context is ambiguous but the word appeared previously in a simpler context.

1.2 Dictionaries

Dictionaries are used to group tokens with specific properties. They associate an entity type to tokens. For NER, several dictionaries were created automatically by pre-processing the training data, according to the following criteria:

- *Dictionary*. Consists in all words annotated as entities that appear more than 5 times in the training corpus with a given type;
- *Prefix*. Three letter prefixes of entity words whose frequency is greater than 9 and whose $\chi^2 > 3.84$.
- *Suffix*. Similarly for suffixes.
- *LastWords*. Words occurring as last in a multi-token entity more than 9 times and whose $\chi^2 > 3.84$.

- *FirstWords*. Similarly for words appearing as first in a multi-token entity.
- *LowerIn*. Lowercase words occurring inside an entity.
- *Bigrams*. All bigrams that precede an entity and occur more than 5 times, whose probability is greater than 0.5 and greater than the probability of their first word.
- *FrequentWords*. Words that occur more than 5 times in the training corpus.
- *Designators*. Words that precede an entity.

The tagger extracts from the dictionaries the following binary features: suffix is present in *Suffix* dictionary; prefix is present in *Prefix* dictionary; token is present in *LastWords*; token is present in *FirstWords*; token is not present in *FrequentWords*; token is present in *LowerIn*.

1.3 Dataset

The dataset was composed of three different corpora:

1. a set of news broadcasts manually transcribed and annotated with Named Entities;
2. the automatic transcription of the same news (without NEs);
3. I-CAB, a corpus of (written) news stories annotated with Named Entities.

Only corpora 1 and 3 contain NEs and could be used for training purposes. These files contain the following information:

- FORM
- PoS (only provided for I-CAB)
- Document-ID
- NE

However corpora 1 and 3 have different origins and are representative of quite different genres: the first one contains manually transcribed spoken broadcast news with no punctuation or sentence boundaries, while the second one is a text corpus composed of news extracted from a local newspaper called “L’Adige”.

Since the test set was composed by broadcast news automatically generated by an automatic speech recognition system (ASR) with no manual correction and with predicted uppercase words, we decided to compute the baseline using only corpus 1, given the closer similarity with the final test set.

For this purpose a basic configuration file was created with no attribute features and with this basic set of local features, which rely only on the words *shape*: the previous word is capitalized; the following word is capitalized; the current word is in upper case; the current word is in mixed case; the current word is a single uppercase character; the current word is a uppercase character and a dot; the current word contains digits; the current word is two digits; the current word is four digits; the current word is made of digits and “/”; the current word contains “$”; the current word contains “%”; the current word contains '; the current word is made of digits and dots.

The baseline was computed training the system on the 90% of the training set and testing it on the remaining 10%; with 100 iterations of the Maximum Entropy algorithm we obtained a F-score of 60.48.

For the tuning process we created different configuration files changing in particular the number of iterations, the value of the *cutoff feature* (an option that prevents the tagger to learn from features appearing a number of times below a specified threshold), the *refine* option (to split the IOB tags into a more refined set) and the attributes features. Moreover we used the Hunpos Tagger [3], trained on the corpus “La Repubblica” [4] to annotate corpus 1 with Part of Speech.

The evaluation was based on a k-fold cross validation, with k = 10. As attributes features for each token we used different combination of the POSTAG, CPOSTAG (first letter of the POSTAG) and NETAG surrounding it. After about 150 tests, we obtained the best results (a F-score of 68.5 on the same development set of the baseline) with the *cutoff* threshold set to 0, the refine feature enabled and the following combination of the attributes features:

Table 1. Attributes features for Run Closed 2

	Run Closed 2
POSTAG	-1 0 1
CPOSTAG	0
NETAG	0

Stanford CRF-Classifier. Due to the peculiarities of this task we decided also to try another tagger based on a different statistical approach: the Stanford Named Entity Recognizer. It is a classifier based on the Conditional Random Fields (CRF) statistical modeling method that uses Gibbs sampling instead of other dynamic programming techniques for inference on sequence models [5]. This tagger works quite well using only the FORM column without any additional information and this can be useful since the system output of the PoS tagger can contain errors.

After the tuning session, two different models were created, one using the full-set of tags in the IOB2 notation (a total of eight classes) and one with only the four semantic classes, i.e. not considering the prefixes ‘B-’ and ‘I-’ From the analysis of the results on the development set we observed that the first model worked better on GPE and LOC, while the previous one on ORG and PER; so we decided to combine the results to improve the performance of the system. The output of this process is Run Closed 1.

I-CAB Corpus. Many experiments were done using as training set also the I-CAB 2009 corpus (~220.000 tokens) in addition to the broadcast news corpus (~40.000 tokens) to give more training examples to the tagger. The basic idea was to use it after removing all punctuation and sentence boundaries to make it more similar the other corpus. The results obtained using both corpora were worst with respect to the ones obtained with only the broadcast news corpus despite its small size, so we decided to produce the final run with the models trained only on the broadcast news corpus. These poor results are probably due to the big difference in the two genres and in particular the meaningfulness of texts in the ICAB corpus with respect to the texts derived from speech by the ASR.

Open Subtask. For the first run of the open subtask we decided to annotate with Super-senses the broadcast news corpus using the Super-sense tagger described in [6] with a model trained on the ISST-SST corpus (~300.000 tokens).

In particular three of the super-senses describe semantic classes similar to the NEs of this task: noun.location (LOC|GPE), noun.person (PER), noun.group (ORG). Hence the basic idea was to exploit super-senses as attributes feature to help the NE tagger to isolate and identify the entities. After some tuning of the features, the best results were obtained on the development set with the same global settings of Run Closed 2 and with the attributes features described in the following table:

Table 2. Attributes features for Run Open 1

	Run Open 1
FORM	0
POSTAG	-2 -1 0 1 2
CPOSTAG	-2 -1
SST	0
NETAG	-2 -1

The second run of the open subtask was created from the output of Run Closed 1 adding some post-processing heuristics. In particular we used a NEs tag dictionary extracted from the corpus itself and ItalWordNet (IWN) [7]. For each capitalized token, the algorithm returns the most common NE tag associated to the token from the self extracted dictionary if available, otherwise it returns the most common super-sense from the IWN dictionary, converted to the corresponding NE tag.

2 Results

The results obtained in the four runs are summarized in the following table.

Table 3. UniPI systems results

	Accuracy	**Precision**	**Recall**	**FB1**
UniPI - run closed 1	95.59%	61.61%	47.23%	53.47
UniPI - run closed 2	95.64%	64.48%	50.45%	56.61
UniPI - run open 1	95.85%	65.90%	52.09%	58.19
UniPI - run open 2	85.45%	54.83%	49.72%	52.15

3 Discussion

The main difficulty of this task derives from the fact that the test set is automatically extracted by the ASR system: it presents many transcription errors, it lacks punctuation, sentence boundaries and capitalization of words is not complete. The test set is therefore quite different form the training set, which was manually revised.

The results obtained on the runs are quite low if we consider the F-score, but the accuracy values are very high. In fact it turns out that the biggest challenge for our systems in this situation was the identification of tokens within the text stream without any marker, like capital letters, to indicate their presence.

The results obtained on the development set (obtained on a portion of the training corpus) were about 15-20 points higher in F-score because all the relevant capital letters were manually added in the corpus. On the test set the heuristic used in Run Closed 2 failed for the same reason: it could be used only for capitalized words.

On the manually corrected test set the results were much higher (see Table 4). This means that our system is weak in dealing with the inaccuracies introduced by the ASR system.

Table 4. UniPI systems results on gold test set

	Accuracy	Precision	Recall	FB1
UniPI - run closed 1	97.64%	78.17%	71.29%	74.57
UniPI - run closed 2	97.14%	74.14%	69.88%	71.95
UniPI - run open 1	97.45%	76.34%	72.75%	74.50
UniPI - run open 2	97.04%	64.90%	70.46%	67.57

References

1. Attardi G., Dei Rossi S., Simi M.: The Tanl Pipeline. In: Proceedings of Workshop on Web Services and Processing Pipelines in HLT, co-located LREC 2010, Malta (2010)
2. Chieu H.L., Ng H.T.: Named Entity Recognition with a Maximum Entropy Approach. In: Proceedings of CoNLL-2003, pp. 160-163. Edmonton, Canada (2003)
3. Halácsy P., Kornai A., Oravecz C.: HunPos – an open source trigram tagger. In: Proceedings of the Demo and Poster Sessions of the 45th Annual Meeting of the ACL, pp. 209–212, Prague, Czech Republic (2007)
4. Attardi G., Simi M.: Overview of the EVALITA 2009 Part-of-Speech Tagging Task. In: Proceedings of Workshop Evalita 2009, ISBN 978-88-903581-1-1 (2009)
5. Finkel J. R., Grenager T., Manning C.: Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In: Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005), pp. 363-370, (2005)
6. Attardi G., Dei Rossi S., Di Pietro G., Lenci A., Montemagni S., Simi M.: A Resource and Tool for Super-sense Tagging of Italian Texts. In: Proceedings of 7th Language Resources and Evaluation Conference (LREC 2010), Malta (2010)
7. Roventini A., Alonge A., Calzolari N., Magnini B., Bertagna F.: ItalWordNet: a Large Semantic Database for Italian. In: Proceedings LREC 2000, Athens (2000)