Training a Dialogue Act Tagger with the μ-TBL System

Torbjörn Lager and **Natalia Zinovjeva**
Department of Linguistics
Uppsala University, SWEDEN
{torbjorn,natalia}@stp.ling.uu.se

ABSTRACT

This short paper describes an attempt to repeat the experiments on dialogue act tagging performed by Ken Samuel and others at the University of Delaware, but with a different learner and a different corpus. We reach an accuracy of 62.1%, which is encouraging.

Keywords

Transformation-based learning, dialogue act tagging

INTRODUCTION

The μ-TBL system – described in detail in [3] – uses the search and database capabilities of the Prolog programming language to implement a generalized form of *transformation-based learning* [1].[1] Through its support of a compositional rule/template formalism and 'pluggable' algorithms, the μ-TBL system can easily be tailored to different learning tasks. In this paper, we will show how the μ-TBL system can be used to train a *dialogue act tagger*. We adopt the approach described by Samuel et al. in [4], and we pick our training- and test data from the Maptask corpus [2].

TRANSFORMATION-BASED LEARNING

The object of transformation-based learning is to learn an ordered sequence of *transformation rules*. Such rules dictate when – based on the context – an utterance should have its tag changed. An example would be "replace the tag for acknowledgement with the tag for yes-reply, if the current utterance contains the word "did", and if the previous utterance is tagged with a yes/no-query tag." Here is how this rule is represented in the rule/template formalism of the μ-TBL system:

```
da:ack>reply_y <-
     u_mem:did@[0] & da:query_yn@[-1]
```

Rules that can be learned in transformation-based learning are instances of *rule templates*. For example, the above rule is an instance of the following template:

```
da:A>B <- u_mem:W@[0] & da:C@[-1].
```

Learning is a matter of repeatedly instantiating rule templates in training data, scoring rules on the basis of counts of positive and negative evidence of them, selecting the highest scoring rule on the basis of this ranking, and applying it to the training data.

THE EXPERIMENTAL SETUP

Corpus Data

We train and test our dialogue act tagger on a subset of the *Maptask corpus* [2]. The Maptask corpus consists of 128 instructional two-person dialogues. One person – the route giver (`g`) – gives another person – the route follower (`f`) – instructions how to navigate a route through a landscape pictured on a map. In the present experiment, we use 35 dialogues (9002 utterances) for training, and 5 dialogues (996 utterances) for testing.

There are 12 dialogue act tags in the Maptask coding scheme: `ready`, `instruct`, `explain`, `check`, `align`, `query_yn`, `query_w`, `ack(nowledge)`, `reply_y`, `reply_n`, `reply_w` and `clarify`. See [2] for a thorough explanation of the acts.

Representation of Corpus Data

The μ-TBL system expects training and test data to be represented as databases of Prolog facts. These are the predicates that we use in the present experiment:

- `u(P,U)` is true iff the utterance `U` is at position `P` in the corpus
- `s(P,S)` is true iff the utterance at position `P` in the corpus was uttered by `S`
- `da(P,A)` is true iff the utterance at position `P` in the corpus is tagged `A`
- `da(A,B,P)` is true iff the utterance at `P` is tagged `A` and the correct tag for the utterance at `P` is `B`

Utterances are represented as lists of words, `S` is either `g` or `f`, and `A` and `B` are tags denoting dialogue acts. Although this representation may seem a bit redundant, it provides exactly the kind of indexing into the data that is needed.

Rule Templates

With an eye to a possible application within a dialogue system, we have decided to make template conditions sensitive to features of the current or previous utterances only. Below, we list the 16 templates that we use.

1. The μ-TBL system – along with corpus data and templates for performing a down-scaled version of the experiment described in here – is available from: http://www.ling.uu.se/~lager/mutbl.html

```
da:A>B <- u_mem:W@[0].
da:A>B <- u_first:W@[0].
da:A>B <- u_last:W@[0].
da:A>B <- u_bigram:W@[0].

da:A>B <- da:C@[-1].
da:A>B <- da:C@[-1,-2].
da:A>B <- da:C@[-1] & da:D@[-2].

da:A>B <- da:C@[-1] & u_mem:W@[0].

da:A>B <- s:C@[0].
da:A>B <- s:C@[0] & da:D@[-1].
da:A>B <- s:C@[0] & u_mem:W@[0].
da:A>B <- s:C@[0] & u_bigram:W@[0].

da:A>B <- s_change:C@[0] & u_mem:W@[0].
da:A>B <- s_change:C@[0] & da:D@[-1].

da:A>B <- u_length:C@[0] & u_mem:W@[0].
da:A>B <- u_length:C@[0] & da:D@[-1].
```

The idea here is that conditions for changing the tag of an utterance are sensitive to the actual words and word combinations used in the utterance, the length of the utterance, the previous dialogue act(s), the speaker's role in the dialogue (giver or follower), and whether the speaker has changed since the previous utterance.

Values of some features can be directly read off from the corpus representation, whereas others are defined in terms of this representation, by means of auxiliary predicates such as these:

```
u_mem(P,W) :- u(P,Ws), member(W,Ws).

u_first(P,W) :- u(P,[W|_]).

s_change(P,no) :-
        s(P,A), P1 is P-1, s(P1,A), !.
s_change(P,yes).
```

EXPERIMENTAL RESULTS

Tagging each utterance in the test data with the most common dialogue act in the training data (`acknowledge`) gave a baseline correctness of 21.6%. The learning process resulted in a sequence of 348 rules, by means of which we were able to tag the test corpus with an accuracy of 62.1%. Thus, our result is not as good as the 75.1% result reported by Samuel et al., but that can probably be explained by the particular characteristics of our training corpus (long and varied dialogues), the small set of templates that we use, and the comparatively short time we have invested in the task.

A CLOSER LOOK AT THE RULES

The system finds many rules where 'cue-words' indicate dialogue acts of various kinds:

```
da:ack>reply_n <- u_mem:'No'@[0]
da:ack>reply_y <- u_first:'Uh-huh'@[0]
da:ack>check <- u_first:'So'@[0]
da:query_yn>instruct <- u_mem:go@[0]
```

It is interesting – and also typical of how transformation rules work – that the second of those rules – which changes an acknowledge tag into a yes-reply tag in the presence of the word "Uh-huh" – is later followed by a rule which reverses that change again if the utterance is preceded by an instruct act:

```
da:reply_y>ack <- da:instruct@[-1] &
                  u_mem:'Uh-huh'@[0]
```

Other rules capture well-known regularities, e.g. the tendency that questions are often followed by replies

```
da:explain>reply_w <-
     s_change:yes@[0] & da:query_w@[-1]
```

or that replies are usually *not* followed by other replies:

```
da:reply_n>ack <- da:reply_n@[-1]
```

Common word order configurations – captured in bigrams – signal other dialogue acts:

```
da:explain>query_yn <-
               u_bigram:(do,you)@[0]
```

Long utterances with pauses in them (transcribed as '...' in the corpus) are usually instructions:

```
da:query_yn>instruct <-
     u_length:'>10'@[0] & u_mem:...@[0]
```

Needless to say, such rules also make a lot of errors, but the errors are often fixed by rules later in the sequence.

SUMMARY AND CONCLUSIONS

Using the μ-TBL system and picking our corpus data from the Maptask corpus, we have repeated the experiments performed by Samuel et al. All in all, we are encouraged by our initial results, and we think we may be able to improve upon them.

ACKNOWLEDGMENTS

We acknowledge the work of the Dialogue Group at the HCRC in Edinburgh in collecting and coding the Maptask corpus, and making it available on the web.

REFERENCES

1. Brill, E., 1995, Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging, *Computational Linguistics*, December 1995.

2. Carletta, J., Isard, A., Isard, S., Kowtko, J.C., Doherty-Sneddon, G., and Anderson, A.H., 1997, The Reliability of a Dialogue Structure Coding Scheme, *Computational Linguistics* 23(1):13-31.

3. Lager, T., 1999, The μ-TBL System: Logic Programming Tools for Transformation-Based Learning, In *Proceeding of the Third International Workshop on Computational Natural Language Learning* (CoNLL-99), Bergen, 1999.

4. Samuel, K., Carberry, S. and Vijay-Shanker, K., 1998, Dialogue Act Tagging with Transformation-Based Learning. In *Proceedings of COLING/ ACL'98*, pp. 1150-1156.