The UMUS system for named entity generation at GREC 2010

Benoit Favre
LIUM, Université du Maine
72000 Le Mans, France
benoit.favre@gmail.com

Bernd Bohnet
Universität Stuttgart
Stuttgart, Germany
bohnet@informatik.uni-stuttgart.de

Abstract

We present the UMUS (Université du Maine/Universität Stuttgart) submission for the NEG task at GREC'10. We refined and tuned our 2009 system but we still rely on predicting generic labels and then choosing from the list of expressions that match those labels. We handled recursive expressions with care by generating specific labels for all the possible embeddings. The resulting system performs at a type accuracy of 0.84 an a string accuracy of 0.81 on the development set.

1 Introduction

The Named Entity Generation (NEG) task consists in choosing a referential expression (complete name, last name, pronoun, possessive pronoun, elision...) for all person entities in a text. Texts are biographies of chefs, composers and inventors from Wikipedia. For each reference, a list of expressions is given from which the system has to choose. This task is challenging because of the following aspects:

1. The data is imperfect as it is a patchwork of multiple authors' writing.
2. The problem is hard to handle with a classifier because text is predicted, not classes.
3. The problem has a complex graph structure.
4. Some decisions are recursive for embedded references, i.e. "his father".
5. Syntactic/semantic features cannot be extracted with a classical parser because the word sequence is latent.

We do not deal with all of these challenges but we try to mitigate their impact. Our system extends our approach for GREC'09 (Favre and Bonhet, 2009). We use a sequence classifier to predict generic labels for the possible expressions.

2 Labels for classification

Each referential expression (REFEX) is given a label consisting of sub-elements:

- The REG08_TYPE as given in the REFEX (name, common, pronoun, empty...)
- The CASE as given in the REFEX (plain, genitive, accusative...)
- If the expression is a pronoun, then one of "he, him, his, who, whom, whose, that", after gender and number normalization.
- "self" if the expression contains "self".
- "short" if the expression is a one-word long name or common name.
- "nesting" if the expression is recursive.

For recursive expressions, a special handling is applied: All possible assignments of the embedded entities are generated with labels corresponding to the concatenation of the involved entities' labels. If the embedding is on the right (left) side of the expression, "right" ("left") is added to the label. Non-sensical labels (i.e. "he father") are not seen in the training data, and therefore not hypothesized.

3 Features

Each reference is characterized with the following features:

- SYNFUNC, SEMCAT, SYNCAT: syntactic function, semantic category, syntactic category, as given in REF node.
- CHANGE, CHANGE+SYNFUNC: previous reference is for a different entity, possibly with syntactic function.
- PREV_GENDER_NUMBER: if the reference is from a different entity, can be "same"

or "different". The attribute is being compared is "male", "female" or "plural", determined by looking at the possible expressions.

- FIRST_TIME: denotes if it's the first time that the entity is seen. For plural entities, the entity is considered new if at least one of the involved entities is new.
- BEG_PARAGRAPH: the first entity of a paragraph.
- {PREV,NEXT}_PUNCT: the punctuation immediately before (after) the entity. Can be "sentence" if the punctuation is one of ".?!", "comma" for ",;", "parenthesis" for "()[]" and "quote".
- {PREV,NEXT}_SENT: whether or not a sentence boundary occurs after (before) the previous (next) reference.
- {PREV,NEXT}_WORD_{1,2}GRAM: corresponding word n-gram. Words are extracted up to the previous/next reference or the start/end of a sentence, with parenthesized content removed. Words are lowercased tokens made of letters and numbers.
- {PREV,NEXT}_TAG: most likely part-of-speech tag for the previous/next word, skipping adverbs.
- {PREV,NEXT}_BE: any form of the verb "to be" is used after (before) the previous (next) reference.
- EMBEDS_PREV: the entity being embedded was referred to just before.
- EMBEDS_ALL_KNOWN: all the entities being embedded have been seen before.

4 Sequence classifier

We rely on Conditional Random Fields[1] (Lafferty et al., 2001) for predicting one label (as defined previously) per reference. We lay the problem as one sequence of decisions per entity to prevent, for instance, the use of the same name twice in a row. Last year, we generated one sequence per document with all entities, but it was less intuitive. To the features extracted for each reference, we add the features of the previous and next reference, according to label unigrams and label bigrams. The c hyperparameter and the frequency cutoff of the classifier are optimized on the dev set. Note that for processing the test set, we added the development data to the training set.

[1] CRF++, http://crfpp.sourceforge.net

5 Text generation

For each reference, the given expressions are ranked by classifier-estimated posterior probability and the best one is used for output. In case multiple expressions have the same labeling (and the same score), we use the longest one and iterate through the list for each subsequent use (useful for repeated common names). If an expression is more than 4 words, it's flagged for not being used a second time (only ad-hoc rule in the system).

6 Results

Evaluation scores for the output are presented in Table 1. The source code of our systems is made available to the community at http://code.google.com/p/icsicrf-grecneg.

Sys.	T.acc	Prec.	Rec.	S.acc	Bleu	Nist
Old	0.826	0.830	0.830	0.786	0.811	5.758
New	**0.844**	0.829	0.816	**0.813**	0.817	6.021

Table 1: Results on the dev set comparing our system from last year (old) to the refined one (new), according to REG08_TYPE accuracy (T.acc), precision and recall, String accuracy (S.acc), BLEU1 an NIST.

About 50% of the errors are caused by the selection of pronouns instead of a name. The selection of the pronoun or name seems to depend on the writing style since a few authors prefer nearly always the name. The misuse of names instead of pronouns is second most error with about 15%. The complex structured named entities are responsible for about 9% of the errors. The selection of the right name such as given name, family name or both seems to be more difficult. The next frequent errors are confusions between pronouns, elisions, common names, and names.

References

Benoit Favre and Bernd Bonhet. 2009. ICSI-CRF: The Generation of References to the Main Subject and Named Entities Using Conditional Random Fields. In *ACL-IJCNLP*.

J. Lafferty, A. McCallum, and F. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Machine Learning*, pages 282–289.