The language component of the FASTY predictive-typing system

Johannes Matiasek
Austrian Research Institute for Artificial Intelligence
johannes.matiasek@ofai.at

Introduction

In this talk we describe the language component of FASTY, a text prediction system designed to improve text input efficiency for disabled users.[1] The FASTY language component is based on state-of-the-art n-gram- based word-level and Part-of-Speech-level prediction and on a number of innovative modules (morphological analysis, collocation-based prediction, compound prediction) that are meant to enhance performance in languages other than English. Together with its modular architecture these novel techniques make it adaptable to a wide range of languages without sacrificing performance. Currently, versions for Dutch, German, French, Italian, and Swedish are supported.

The Basic Prediction Model

The core prediction engine is based on a statistical language model, as this is the case with most current predictive typing system (see, e.g. Carlberger (1998)). The FASTY language model utilizes word unigrams, word bigrams, PoS trigrams and the probability distribution $P(t|w)$, i.e. the probability that PoS tag t occurs with given word w. The PoS statistics have been collected from tagged corpora, the word statistics have been obtained from untagged corpora[2]. The connection between tags and wordforms is established via a large[3], FST-based, morphosyntactic lexicon containing the admissible tags for every wordform. $P(t|w)$ is approximated by normalizing the overall distribution of the admissible tags for w.

The word model estimates the next word by interpolating the bigram based estimate (given the previous word) and the unigram based estimate. For both estimates the already entered prefix of the current word has to be taken into account. In order to do this efficiently, *trie* datastructures are used in FASTY to associate frequency information with single words and word n-grams. An example of such a trie can be seen in Fig. 1. In order to calculate the relative frequency of a word w one has to lookup the count of the node at the end of the path labeled with the character sequence of w, subtract the counts of the target nodes of the outgoing arcs, and divide it by the count of the node associated with the already entered prefix.

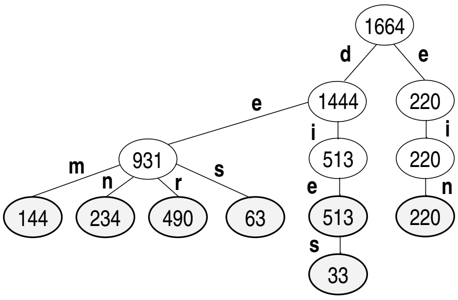

Figure 1: Sample Trie

For modeling PoS tag sequences a second order, i.e. a trigram-based, Markov model is used, combining the PoS-tag estimates of the previous and the pre-previous word to yield an estimate of the PoS-tag for the current word. Based on the tag model, terms weighing the unigram and bigram based word frequencies with a factor indicating the appropriateness of that word in the current syntactic context are added to the interpolated word model.[4]

Whenever the word model is low on possible predictions, the morphological lexicon is consulted as an additional resource, and words compatible with the already entered prefix are added with the probability of a hapax.

In order to adapt to specific user needs, the word model is supplemented by a *User Dictionary* that is collected from user texts (or texts fitting in the user's genre) in the same way as the general word model. The user dictionary is interpolated with the general dictionary with adjustable weights, and it may be automatically incremented during the operation of the system. Furthermore, a user-customizable *Abbreviation Dictionary* may be used for frequent words or phrases.

Model Extensions

The main goal of the FASTY project was to develop a text prediction system that could be adapted to most European languages delivering a KSR comparable to English language systems, which - due to grammatical and orthographical peculiarities of English - did not perform as well when applied to other languages.

Compound Prediction

Compounding is a common and often very productive cross-linguistic mean to form complex words. In many languages compounds are commonly written as single orthographic words. Since the performance of a predictive-typing system crucially depends on the lexicon, and, on the other hand, compounding leads to a considerable amount of orthographic words that cannot, even in principle, be listed in a lexicon, a dedicated module for predicting compounds has been developed.

[1] FASTY (Faster Typing for Disabled Persons) was an EU-sponsored project (IST-2000-25420) in FP5 with consortium partners from five countries. The project was finished in 2004. An offspring of that project is the EMU system that employs an improved user-interface and is based on the same language component. Demo versions can be downloaded from `http://www.is.tuwien.ac.at/emu/index.html`

[2] for German, a 28 million word corpus of Austrian newswire text has been used

[3] The current lexicon for German contains 568536 wordforms with more than 2 million readings.

[4] A more formal treatment can be found in Trost et al. (2005)

Dagstuhl Seminar Proceedings 05382
Efficient Text Entry
http://drops.dagstuhl.de/opus/volltexte/2006/516

Based on our analysis of the frequency, productivity and structural properties of German compounds, we constructed a model[5] in which we try to predict N+N compounds by treating them as the sequence of a modifier and a head and by relying on the distributional properties of modifiers and heads as independent units in the training corpus. Thus, modifier and head are predicted separately in two steps: Modifier prediction is in the same way as for single words, however, if a predicted word is more likely to appear as the first part of a compound than as single word, no space is added after accepting the word (otherwise, the user has to backspace to enter compound prediction mode). Compound head prediction takes into account the

- unigram probability of candidate head
- bigram probability of candidate head (based on the word preceding the whole compound!)
- likelihood of candidate head to occur in compound head position
- semantic class-based modifier/head-bigram probability of candidate head[6]

Collocation-based Prediction

We expected to get improved prediction when taking semantic/topical dependencies among words into account. In order to exploit these dependencies, we collected trigger-target pairs from corpus using mutual information as association measure. For prediction, every trigger is associated with its targets by an integer score indicating the association strength. During the prediction process, the collocation-based model keeps track of the last n words entered and maintains a dynamic trie containing the trigger-targets of these words. The score if a target word entered into that trie decays with the distance of its trigger to the current word. Using this dynamic trie (that has to be reconstructed every time the context window changes) predictions of semantically plausible words can be made easily (for a detailed account on this approach cf. Matiasek & Baroni (2003)).

Evaluating the collocation-based prediction module, it turned out that there was a consistent, but very moderate improvement of KSR. Given the big resources and rather expensive computation needed, this module is at the moment not part of the FASTY language component.

Grammar-based Prediction

Experiments have been made to identify apparently ungrammatical predictions by means of partial parsing, and to penalize them. Although at the first glance results seemed promising, an overall evaluation showed that the KSR improvements were only marginal. Given the considerable overhead and portability difficulties induced by the parsing module, this enhancement was not included in the final system (see Gustavii & Pettersson (2003)).

Evaluation

Numerous simulation runs to compute the KSR, varying language and parameter settings, have been performed on texts from different genres. The size of the prediction window is the most influential parameter on the achievable KSR.

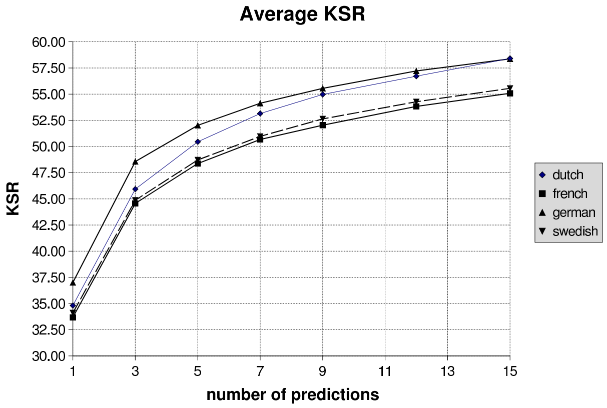

Figure 2: KSR vs. Prediction Window Size

It turned out that the KSR for texts from genres closer to the training corpus was consistently higher. Thus, for optimum results, a appropriately trained user dictionary is of utmost importance.

Future Perspectives

An issue that has not been explored within the FASTY project is the use of reduced, ambigous keyboards. The language component, however, is ready for ambiguous character mappings. Both, the trie module used for storing frequency dictionaries (unigrams and bigrams), and the FST module used for storing the morphological lexion, are able to cope with table-defined character mappings associating one input character with one (or more) characters of the words/phrases stored in the dictionary. Thus, the whole prediction machinery as described above can be used, without additional effort, to assign the most probable reading to an ambigous input sequence, and even may be able to achieve keystroke savings in spite of a reduced keyboard. A version of the language component demonstrating that aspect will be presented at the workshop. An unresolved problem, however, remains: entering out-of-vocabulary words with a reduced keyboard is currently not possible (no such problem occurs, of course, with a full keyboard).

References

Baroni M., J. Matiasek, and H. Trost. 2002. Wordform- and Class-Based Prediction of the Components of German Nominal Compounds in an AAC System. in Tseng S.-C.(ed.), *COLING 2002*. Taipei, Taiwan. pp.57-63.

Carlberger J. 1998. *Design and Implementation of a Probabilistic Word Prediction Program*. Royal Institute of Technology (KTH).

Gustavii E. and E. Pettersson E. 2003. *A Swedish Grammar for Word Prediction*. Master's Thesis. University of Uppsala.

Matiasek J. and M. Baroni. 2003. Exploiting Long Distance Collocational Relations in Predictive Typing, in *Proceedings of the EACL Workshop on Language Modeling for Text Entry Methods*. Budapest. Hungary. pp.1-8.

Trost H., J. Matiasek, and M. Baroni. 2005. The Language Component of the FASTY Text Prediction System. *Applied Artificial Intelligence*. 19**(8)**, pp. 743-781.

[5] The same model has been successfully used for Dutch and Swedish.

[6] Semantic classes have been collected from corpus by clustering, using mutual information as an association measure. Replacing modifiers and heads of compounds in the corpus by their class-ids and counting them yields the class-based bigrams (see [Baroni et al. (2002)).