N-Gram Language Modeling for Robust Multi-Lingual Document Classification

Jörg Steffen

German Research Center for Artificial Intelligence GmbH
Stuhlsatzenhausweg 3, D-66123 Saarbrücken, Germany
steffen@dfki.de

Abstract

Statistical n-gram language modeling is used in many domains like speech recognition, language identification, machine translation, character recognition and topic classification. Most language modeling approaches work on n-grams of terms. This paper reports about ongoing research in the MEMPHIS project which employs models based on character-level n-grams instead of term n-grams. The models are used for the multi-lingual classification of documents according to the topics of the MEMPHIS domains. We present methods capable of dealing robustly with large vocabularies and informal, erroneous texts in different languages. We also report on our results of using multi-lingual language models and experimenting with different classification parameters like smoothing techniques and n-grams lengths.

1. Introduction

The MEMPHIS project[1] aims at developing a platform for premium content services targeting mobile users and mobile devices. The system collects multi-lingual content from various sources, merges it, extracts informations from it and optionally summarizes and translates it. Users can subscribe to a service choosing information topics within a domain and preferred target devices and output formats. One central task in this scenario is the reliable multi-lingual classification of acquired content according to the service topics.

One of the domains in MEMPHIS are book announcements released by various book shops and publishers on their internet sites. Our goal is to automatically assign topics to these announcements within the fixed set of topics defined in MEMPHIS. The domain of book announcements yields several issues that must be considered when doing classification: book announcements are rather informal texts with an open-ended vocabulary. Additionally, on book shop sites like Amazon.com, book announcements may also contain user reviews with spelling mistakes and/or no case distinction. The classification approach must be robust enough to handle this. Another issue is the multi-linguality of the acquired book announcements. They must be classified using a set of topics that is defined independently of the language. Finally, some topics might overlap so that there is content to which more than one topic must be assigned.

This paper describes our ongoing research in the MEMPHIS project to implement a classifier that can handle all these issues.

2. Classification Fundamentals

Term-based classification approaches have some disadvantages: They need a linguistic preprocessing step that at least identifies the terms. Additionally, they suffer from the sparse data problem: Even with large training corpora, there will be a significant number of terms in test data that are not contained in the training data. This problem becomes worse within the wide vocabulary range of the book announcement domain. A common way to decrease sparse data is stemming. This requires an additional expensive linguistic preprocessing. Another problem are terms with spelling mistakes that can't be reduced to a stem, so they extend the vocabulary range even more.

2.1. Character-Level N-Gram Modeling

In MEMPHIS we use a classification approach based on character-level n-grams. They are created by splitting a text into overlapping character sequences of length n, treating all non-whitespace and whitespace characters equally, which means that word borders, punctuation, etc. may appear within an n-gram. This approach needs no linguistic preprocessing at all and is completely language independent. It is also very robust when working on "noisy" texts with spelling errors, since a spelling error influences only the n-grams derived from its immediate neighborhood. In addition, using character-level n-grams results in less sparse data, because there are far less possible n-grams than there are possible terms. The good performance of this classification approach has been show recently in (Peng et al., 2003).

2.2. Model Training and Classification

Common techniques for machine learning that have been adapted for automatic topic classification include naive Bayes, Rocchio, k-nearest neighbors, support vector machines and maximum entropy. For an overview see (Sebastiani, 2002). The classification approach in MEMPHIS uses character-level n-grams with the naive-Bayes classifiers. A document is treated as a character sequence $s = c_1, ..., c_n$. We denote this character sequence as c_1^n. Before it can be classified, language models must be trained. This requires a training corpus with example documents that are tagged with one or more topics. For each topic, a statistical model is created. This makes it easy to extend or reduce the topics the classifier covers. To assign a topic to a document, we calculate the probability of its character sequence for each language model. The result is a ranking of topics, and we pick the top ranking topic as topic of the document. The probability of a document's character sequence can be

[1] http://www.ist-memphis.org

expressed with the chain rule of probability as

$$P(s) = P(c_1^n) = \prod_{i=1}^{n} P(c_i|c_1^{i-1}) \quad (1)$$

A common approximation in n-gram language models is that the probability of a character depends only on the preceding n - 1 characters. This means

$$\prod_{i=1}^{n} P(c_i|c_1^{i-1}) \approx \prod_{i=1}^{n} P(c_i|c_{i-n+1}^{i-1}) \quad (2)$$

A language model contains conditional probabilities of character-level n-grams based on their frequencies counted in the example documents of the training corpus. The most straightforward approach is the *maximum likelihood estimate*.

$$P(c_i|c_{i-n+1}^{i-1}) = \frac{\#(c_{i-n+1}^{i})}{\#(c_{i-n+1}^{i-1})} \quad (3)$$

This approach is not applicable in that form since it can't handle sparse data in a proper way.

2.3. Sparse Data and Smoothing

The sparse data problem, as mentioned above for term-based classification approaches, remains to be handled when using character-level n-grams, too. With the maximum likelihood estimation, the probability of an n-gram that was unseen in the training examples would be zero. To avoid this, we must reduce the probability of known n-grams to reserve some room in the probability space for unknown n-grams. This procedure is called *smoothing*. A simple approach is to pretend that each n-gram occurs more often than it actually does. More advanced approaches "discount" the number of n-grams in the numerator of (3) and then derive the probability for unknown n-grams from lower-order model, e.g. for an unknown trigram, the probability is derived from the bigram model. Language models that use this technique are called *backoff models*. The probability of an n-gram in such models is calculated as

$$P_{bo}(c_i|c_{i-n+1}^{i-1}) = \quad (4)$$
$$\begin{cases} P(c_i|c_{i-n+1}^{i-1}) & if\ \#(c_{i-n+1}^{i}) > 0 \\ \gamma(c_{i-n+1}^{i-1}) \times P_{bo}(c_i|c_{i-n+2}^{i-1}) & if\ \#(c_{i-n+1}^{i}) = 0 \end{cases}$$

where $\gamma(c_{i-n+1}^{i-1})$ is a scaling factor that ensures that the probabilities sum up to one. *Interpolated models* are an extended variant, where the lower-model probability is not only incorporated in the probability of unknown n-grams, but also for known n-grams. In section 3.2. we will describe the smoothing techniques used in the MEMPHIS classifier.

3. Document Classification in MEMPHIS

The task of the document classification in MEMPHIS is to automatically assign topics to book announcements acquired from the internet sites of book shops and publishers. The seven topics covered in MEMPHIS are *biography*, *film*, *food*, *health*, *music*, *sports* and *travel*.

3.1. Linguistic Resources

To train language models and to test the classification performance, we acquired English and German corpora from several internet sources. Since the topics structure differs considerably among the different sites, we defined a mapping from each MEMPHIS topic to one or more topics or subtopics of the specific site. If the same book was acquired for more than one MEMPHIS topic, it was labeled with multiple topics.

Amazon offers a web service[2] that allows to retrieve book offers for a specified topic in XML format. This makes it easy to extract only the content of a book announcement that contains informations relevant for classification, namely the book title, a description and user reviews. The remaining information is ignored. We acquired a German and an English corpus from Amazon. Each corpus is balanced, meaning that we have the same number of document (1000) per topic. Each corpus contains about 18% of multi-labeled documents. The English corpus has a size of 13 MB, the German corpus has a size of 10 MB.

We also acquired some smaller unbalanced corpora from Bol.de (1279 docs, 1 MB), Buecher.de (2348 docs, 2 MB), Powells.com (8266 docs, 7 MB) and Randomhouse.com (3026 docs, 4 MB). These corpora contain 10% - 15% multi-label documents. These documents are acquired in HTML format. To extract the classification relevant content, we use the MEMPHIS extraction component (Kasper et al., 2004).

3.2. The MEMPHIS Classifier

We implemented the MEMPHIS classifier in a way that allows us to test its performance using different classification parameters. The most obvious parameter is the n-gram length, where we experimented with lengths from 2 to 5.

We also implemented a number of smoothing techniques that result in both backoff and interpolated language models: a simple backoff approach based on (Katz, 1987), a backoff approach that uses the regression analysis of Simple Good-Turing Smoothing (Gale and Sampson, 1996) to estimate the n-gram discounts, a backoff and an interpolated version of Absolute Smoothing (Ney et al., 1994), the interpolated approach of Kneser-Ney Smoothing (Kneser and Ney, 1995) and finally a variation of Kneser-Ney Smoothing as described in (Chen and Goodman, 1998) (referred to as Chen-Goodman Smoothing in the following). Some of these smoothing techniques include parameter estimations that had to be adapted for the "frequency of frequencies" distribution of character-level n-grams, because this distribution is different from the Zipfian distribution of term n-grams.

Finally, we implemented an optional non-linguistic preprocessing step where we strip all whitespaces from a document and convert all characters to lower case. To preserve the information of word borders, the first character following one or more removed whitespaces is always converted to upper case. For clarification, see the following original and converted sequence:

LIFE STORIES: Profiles from the New Yorker
LifeStories:ProfilesFromTheNewYorker

[2] http://www.amazon.com/webservices

	KATZ	GOOD-TURING	ABS-IP	ABS-BO	KNESER-NEY	CHEN-GOODMAN
2-grams	0.7786	0.80686	0.8087	0.8075	**0.80988**	0.80943
3-grams	0.87404	0.886	0.88671	0.88719	**0.88904**	0.88861
4-grams	0.897	0.90492	0.90069	0.90679	0.90716	**0.90717**
5-grams	0.8959	0.89704	0.8959	**0.90279**	0.9	0.90022

Table 1: Average F_1 values for German Amazon corpus

	KATZ	GOOD-TURING	ABS-IP	ABS-BO	KNESER-NEY	CHEN-GOODMAN
2-grams	0.80517	0.8402	0.84041	0.84071	**0.84169**	0.84161
3-grams	0.89918	0.9103	0.90699	0.91041	0.91116	**0.91144**
4-grams	0.91673	0.92221	0.91859	0.92324	**0.92465**	0.92458
5-grams	0.91283	0.91585	0.91382	0.91864	0.91862	**0.91868**

Table 2: Average F_1 values for English Amazon corpus

4. Evaluation

In the following, we describe our experiments and interpret the results using different combinations of classification parameters with mono-lingual and multi-lingual language models. We also do a cross-site classification evaluation and examine the quality of the topic ranking returned by the classifier for each document.

Classification effectiveness is measured in terms of precision and recall defined as

$$Precision = \frac{true\ positives}{true\ positives + false\ positives} \quad (5)$$

$$Recall = \frac{true\ positives}{true\ positives + false\ negatives} \quad (6)$$

The overall precision and recall across all topics is then calculated using macro-averaging. Both values are then combined using the f-measure

$$F_1 = \frac{2 \times precision \times recall}{precision + recall} \quad (7)$$

In the experiments concerning different classification parameters and cross-site evaluation, the classifier assigns each document a single topic. If a document is originally labeled with multiple topics and the topic assigned by the classifier is among them, we consider this as a true positive. The complete set of topics assigned to a document is considered in the rank evaluation in 4.4.

4.1. Mono-Lingual Language Models

We examined the influence of the smoothing technique and the n-gram length on the classification performance in mono-lingual language modeling. We randomly split the German and English Amazon corpus into a training part of 80% and a testing part of 20% and measured the classification performance for each combination of smoothing technique and n-gram length. Tables 1 and 2 show the average f-measure for a series of 10 random splits. The first thing to notice is that for a fixed n-gram length, the performances for the different smoothing techniques are very close together, although Kneser-Ney and Chen-Goodman smoothing usually score best and Katz smoothing always has the worst performance.

Among all smoothing techniques, the performance increases as the length of the n-grams grows, but only up to a length of 4. The performance slightly drops with models using 5-grams. The reason for this is that the corpora are not large enough to provide reliable counts for 5-grams. But even with larger corpora available, we would stick with 4-grams since the expected increase in performance using 5-grams would be minimal and the 5-gram models are much larger than 4-gram models.[3]

4.2. Multi-Lingual Language Models

We repeated the experiments describe in 4.1., but with a corpus composed of both the English and the German Amazon documents, so we now have twice as much training and testing documents. The results are shown in table 3. Now, 5-gram models perform slightly better than the 4-gram models, as predicated above. We also observe that with growing n-gram length the performance comes close to the one of the mono-lingual models, and in some cases it's even better. We conclude that for n-grams of length 4 and 5, the "noise" introduced by mixing the models is compensated by the increased training corpus size.

4.3. Cross-Site Evaluation

We examined how the classification performance changes when the documents to classify come from a different site than the ones used to train the language models. We used the complete English and German Amazon corpora to train an English, a German and a mixed language model[4] and used these models to classify the non-Amazon corpora. The results are show in table 4. The performance is still very good, although slightly worse than with Amazon test documents. This shows that each site has a certain "style" that influences the classification. We also observe

[3] For the Amazon corpora, a 5-gram model is about 3,5 times as large as a 4-gram model.

[4] We used Chen-Goodman smoothing and 4-grams.

	KATZ	GOOD-TURING	ABS-IP	ABS-BO	KNESER-NEY	CHEN-GOODMAN
2-grams	0.4973	0.50923	0.50844	0.51224	0.51395	**0.51421**
3-grams	0.79259	0.79709	0.79709	0.79918	**0.80273**	0.80253
4-grams	0.89156	0.8932	0.89316	0.89763	**0.90132**	0.9013
5-grams	0.89966	0.90035	0.90104	0.90729	0.90906	**0.90923**

Table 3: Average F_1 values for mixed Amazon corpus

again that the results with the double sized multi-lingual mixed model are as good as with the mono-lingual ones.

	English	German	Mixed
Powells.com	0.8633	———	0.8631
Randomhouse.com	0.8448	———	0.8432
Bol.de	———	0.8981	0.9056
Buecher.de	———	0.7978	0.8238

Table 4: F_1 values for cross-site evaluation

4.4. Ranking Evaluation

For each document, the classifier returns a ranking of topics. The ranking quality can be measured using the *11 point average precision*. It is based on how far one has to go down the ranking to find all topics originally assigned to the document. We repeated the experiments of 4.1. and 4.2. for Chen-Goodman smoothing and calculated the 11 point average precision. The results are shown in table 5. We observe a good performance that behaves the same way as the f-measure concerning n-gram length and model size. In future research we will use the ranking as a base for multi-label classification.

	English	German	Mixed
2-grams	0.8101	0.79374	0.56955
3-grams	0.84633	0.83637	0.78325
4-grams	0.85345	0.84204	0.83744
5-grams	0.84969	0.83751	0.84176

Table 5: 11 point average precision for language models using Chen-Goodman smoothing

4.5. Whitespace Stripping

In all experiments described above, we used the whitespace stripping described in 3.2. This yielded in some cases an improvement of the average f-measure of up to 0.05, but it varies with the n-gram length and the size of the training data. A disadvantage of the whitespace stripping is that the number of different n-grams is increased since more n-grams contain characters from adjacent terms. This leads to the larger models. It is subject of further research to find the circumstances under which whitespace stripping yields the best performance boost and when the trade-off between improved performance and larger model sizes is useful.

5. Conclusions and Outlook

We described a robust multi-lingual classification approach using character-level n-grams that performs very well in assigning topics to informal, erroneous texts of the book announcement domain. Future research will try to extend the approach in a way that allows the automatic assigning of multiple topics to a document. A possibility would be to learn thresholds values based on the corresponding probabilities of the topics ranking.

6. Acknowledgments

The work reported here is part of the *MEMPHIS* project funded by the European Commission under contract IST-2000-25045 in the *Information Society Technologies* (IST) program.

7. References

Chen, Stanley F. and Joshua Goodman, 1998. An empirical study of smoothing techniques for language modeling. Technical Report 10-98, Harvard University.

Gale, William A. and Geoffrey Sampson, 1996. Good-turing frequency estimation without tears. Cognitive Science Research Paper 407, University of Sussex.

Kasper, Walter, Jörg Steffen, Jakub Piskorski, and Paul Buitelaar, 2004. Integrated language technologies for multilingual information services in the memphis project. In *The 4th International Conference on Language Resources and Evaluation (LREC2004)*. Paris, France: ELRA - European Language Resources Association. To appear.

Katz, Slava M., 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(1):400–401.

Kneser, Reinhard and Hermann Ney, 1995. Improved backing-off for m-gram language modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1.

Ney, Hermann, Ute Essen, and Reinhard Kneser, 1994. On structuring probabilistic dependences in stochastic language modelling. *Computer Speech and Language*, 8:1–38.

Peng, Fuchun, Dale Schuurmans, and Shaojun Wang, 2003. Language and task independent text categorization with simple language models. In Marti Hearst and Mari Ostendorf (eds.), *HLT-NAACL 2003: Main Proceedings*. Edmonton, Alberta, Canada: Association for Computational Linguistics.

Sebastiani, Fabrizio, 2002. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47.