Modified Kneser-Ney Smoothing of n-gram Models

Frankie James

RIACS Technical Report 00.07

October 2000

Modified Kneser-Ney Smoothing of n-gram Models

Frankie James

This report examines a series of tests that were performed on variations of the modified Kneser–Ney smoothing model outlined in a study by Chen and Goodman. [2] We explore several different ways of choosing and setting the discounting parameters, as well as the exclusion of singleton contexts at various levels of the model.

Statistical language modeling can be used effectively to provide a baseline for recognition accuracy when studying other forms of speech and language recognition. Perplexities computed using smoothed n-gram models can later be compared to language models based on grammars. In this paper, we look at perplexities calculated on ATIS travel data using the statistical language model known as modified Kneser–Ney. We explore four variations of the basic algorithm outlined in Chen and Goodman [2], and select one that appears to perform significantly better on our test data. We plan to use this model as the baseline for our analysis of future grammar models.

The modified Kneser–Ney algorithm is an extension of Kneser and Ney's algorithm introduced in 1995 [3], which itself is an extension of absolute discounting. Like absolute discounting, the Kneser–Ney algorithm calculates the probability of a word following a particular context by computing the raw probability of the word following the context and subtracting a discounting amount. This discounting amount is then re-added equally to all n-gram probabilities having the same context, by means of a multiplicative factor that is combined with the probability of the word in the next lower level of the model. That is, the discounted raw probability of the n-gram is linearly interpolated with the smoothed probability of the (n–1)-gram created by removing the first word of the context. In absolute discounting, the lower level probability is calculated in the same way as the higher level. However, in Kneser–Ney smoothing, the lower level probability is a smoothed probability calculated *not* by computing the raw probability of the word following the context, but by computing the number of different contexts that the word follows in the lower order model. The modified Kneser–Ney algorithm is further extended by using three discounting parameters (that, in the highest order model at least, are based on the number of occurrences of the n-gram) instead of the single parameter used in standard Kneser–Ney smoothing and absolute discounting.

Models Used

The modified Kneser–Ney algorithm, as presented in Chen and Goodman [2], is not completely specified. The algorithm leaves open to interpretation both the selection and initialization of the discounting parameters. In this section, we will present a set of four modifications which we implemented, and the equations for their calculation.

There are (at least) two possible ways to select the discounting parameters used in the modified Kneser–Ney smoothing algorithm: based on the n-gram count, and based on number of extended contexts of the n-gram. Additionally, it is possible to use different methods to *set* the discounting parameters, which, based on work by Ries (cited in [2]), are calculated according to the number of n-grams that appear one, two, three, or four times. Finally, modified Kneser–Ney smoothing can be implemented so that singleton contexts are excluded from any or all levels of the model.

All of the models described in this section are based on the algorithm presented in Chen and Goodman. In this algorithm, each order of the model is calculated by interpolating between a raw probability for the n-gram and a smoothed probability for the (n–1)-gram. In addition, there are different models for the highest order and lower orders of n. For the highest order model, the equation is as follows:

$$p(w_i|w_{i-n+1}^{i-1}) = \frac{c(w_{i-n+1}^{i}) - D(c(w_{i-n+1}^{i}))}{\sum_{w_i} c(w_{i-n+1}^{i})} + \gamma(w_{i-n+1}^{i-1})\, p(w_i|w_{i-n+2}^{i-1})$$

$$\text{where } D(c) = \begin{cases} 0 \text{ if } c = 0 \\ D_1 \text{ if } c = 1 \\ D_2 \text{ if } c = 2 \\ D_{3+} \text{ if } c \geq 3 \end{cases}$$

and

$$\gamma(w_{i-n+1}^{i-1}) = \frac{D_1 N_1(w_{i-n+1}^{i-1}\bullet) + D_2 N_2(w_{i-n+1}^{i-1}\bullet) + D_{3+} N_{3+}(w_{i-n+1}^{i-1}\bullet)}{\sum_{w_i} c(w_{i-n+1}^{i})}$$

The D values for the model are calculated using the following equations:

$$Y = \frac{n_1}{n_1 + 2n_2}$$

$$D_1 = 1 - 2Y\frac{n_2}{n_1}$$

$$D_2 = 2 - 3Y\frac{n_3}{n_2}$$

$$D_{3+} = 3 - 4Y\frac{n_4}{n_3}$$

where n_1 is the number of n-grams that appear exactly once, n_2 is the number of n-grams that appear exactly twice, etc.

MODKN–COUNT

The first modification on Kneser–Ney that we tested was modkn–count. In this model, we chose to select the discounting parameters in the lower order models based on the count of the n-gram in question, as in the highest order model. In this case, the equation for the smoothing model is as follows:

$$p(w_i|w_{i-n+1}^{i-1}) =$$

$$\frac{\max\left\{N_{1+}(\bullet w_{i-n+1}^{i}) - D(c(w_{i-n+1}^{i})), 0\right\}}{\sum_{w_i} N_{1+}(\bullet w_{i-n+1}^{i})} + \gamma(w_{i-n+1}^{i-1})p(w_i|w_{i-n+2}^{i-1})$$

Then, to make the probabilities add up to one, we need to calculate γ by taking into account any (raw) probabilities that are set to zero:

$$\gamma(w_{i-n+1}^{i-1}) =$$

$$\frac{D_1M_1(w_{i-n+1}^{i-1}\bullet) + D_2M_2(w_{i-n+1}^{i-1}\bullet) + D_{3+}M_{3+}(w_{i-n+1}^{i-1}\bullet) + \sum_{w_i} Z_{1+}(\bullet w_{i-n+1}^{i})}{\sum_{w_i} N_{1+}(\bullet w_{i-n+1}^{i})}$$

where

$$M_j(w_{i-n+1}^{i-1}\bullet) = \left|\left\{w_i : \left(c(w_{i-n+1}^{i}) = j \wedge N_{1+}(\bullet w_{i-n+1}^{i}) \geq D(c(w_{i-n+1}^{i}))\right)\right\}\right|$$

and

$$Z_{1+}(\bullet w_{i-n+1}^{i}) = \left|\left\{x : c(xw_{i-n+1}^{i}) > 0 \wedge N_{1+}(\bullet w_{i-n+1}^{i}) < D(c(w_{i-n+1}^{i}))\right\}\right|$$

MODKN–EXTEND

As we can see in the equations presented above, the lower order models for modified Kneser–Ney use the number of extended context of the n-gram in question as the metric in both the numerator and denominator. Therefore, it seems more logical to select the discounting parameters based on this number, rather than n-gram count. Indeed, this is the method that Chen and Goodman used in their own tests, which we call modkn–extend. In this case, the equation for the lower order model becomes:

$$p(w_i|w_{i-n+1}^{i-1}) =$$

$$\frac{\max\left\{N_{1+}(\bullet w_{i-n+1}^{i}) - D(N_{1+}(\bullet w_{i-n+1}^{i})), 0\right\}}{\sum_{w_i} N_{1+}(\bullet w_{i-n+1}^{i})} + \gamma(w_{i-n+1}^{i-1})\, p(w_i|w_{i-n+2}^{i-1})$$

where

$$\gamma(w_{i-n+1}^{i-1}) = \frac{\sum_{w_i} D(N_{1+}(\bullet w_{i-n+1}^{i}))}{\sum_{w_i} N_{1+}(\bullet w_{i-n+1}^{i})}$$

MODKN–DIFFD

In setting the discounting parameters, Chen and Goodman used a set of equations presented as a personal communication from Ries. These equations are based on the frequency of n-grams with one, two, three, and four counts. However, if we choose the discounting parameter used in the lower order models based on the number of extended contexts of the n-gram (as in modkn–extend), there is a clear alternative to setting the D parameters based on these frequencies. In this case, we can set the D parameters based on the number of different extended contexts that occur one, two, three or four times.

The equations for this model (which we call modkn–diffd) are identical to those for modkn–extend, except for the calculation of the D parameters. In this model, the equations for calculating the D parameters are:

$$Y = \frac{e_1}{e_1 + 2e_2}$$

$$D_1 = 1 - 2Y\frac{e_2}{e_1}$$

$$D_2 = 2 - 3Y\frac{e_3}{e_2}$$

$$D_3 = 3 - 4Y\frac{e_4}{e_3}$$

where

$$e_j = \left|\{w_1 \ldots w_n : \left|\{x : xw_1 \ldots w_n\}\right| = j\}\right|$$

MODKN–FLEX

The final modification we explored was the elimination of singleton contexts from any or all levels of the model. It is expected that this technique will allow the smoothing to work better since we are not giving too much weight to contexts and n-grams that only appear once in the training data. Our code allowed us to choose to eliminate singleton contexts from all levels of the model above a set threshold. Our tests used thresholds of 2, 3, 4, 5, 6, 7, and 8. For any levels below the threshold, the standard equations from modkn–extend are used. For levels above or equal to the threshold, the singleton contexts were eliminated using the equations listed below.

For the highest order model, we have:

$$p(w_i|w_{i-n+1}^{i-1}) = p_{raw}(w_i|w_{i-n+1}^{i-1}) + \gamma(w_{i-n+1}^{i-1})p(w_i|w_{i-n+2}^{i-1})$$

where

$$p_{raw}(w_i|w_{i-n+1}^{i-1}) = \begin{cases} \dfrac{C(w_{i-n+1}^{i}) - D(C(w_{i-n+1}^{i}))}{\sum_{w_i} C(w_{i-n+1}^{i})} & \text{if } C(w_{i-n+1}^{i-1}) > 1 \\ 0 & \text{otherwise} \end{cases}$$

and

$$\gamma(w_{i-n+1}^{i-1}) = \frac{\sum_{w_i \text{ s.t. } C(w_{i-n+1}^{i-1}) > 1} D(C(w_{i-n+1}^{i})) + \sum_{w_i \text{ s.t. } C(w_{i-n+1}^{i-1}) = 1} 1}{\sum_{w_i} C(w_{i-n+1}^{i})}$$

For the lower order models, then, we have:

$$p(w_i|w_{i-n+1}^{i-1}) = \frac{NUM(w_{i-n+1}^{i})}{\sum_{w_i} N_{1+}(\bullet w_{i-n+1}^{i})} + \frac{\sum_{w_i} DISC(w_{i-n+1}^{i})}{\sum_{w_i} N_{1+}(\bullet w_{i-n+1}^{i})} p(w_i|w_{i-n+2}^{i-1})$$

where

$$NUM(w_{i-n+1}^{i}) = \begin{cases} max\left\{N_{1+}(\bullet w_{i-n+1}^{i}) - D(N_{1+}(\bullet w_{i-n+1}^{i})), 0\right\}, \\ \qquad \text{if } C(w_{i-n+1}^{i-1}) > 1 \\ 0 \text{ otherwise} \end{cases}$$

and

$$DISC(w_{i-n+1}^{i}) =$$

$$\begin{cases} D(N_{1+}(\bullet w_{i-n+1}^{i})), \\ \quad \text{if } \left(C(w_{i-n+1}^{i-1}) > 1 \wedge N_{1+}(\bullet w_{i-n+1}^{i}) > D(N_{1+}(\bullet w_{i-n+1}^{i}))\right) \\ N_{1+}(\bullet w_{i-n+1}^{i}), \text{ otherwise} \end{cases}$$

Statistical Comparisons of Data

For each of the modifications to the modified Kneser–Ney algorithm listed above, we performed model training on a corpus by removing one tenth of the corpus and setting it aside for testing. This test was repeated ten times, so that on each run, a different segment was left out. After ten models were trained per algorithm, the perplexities of the models were computed using two different data test sets: (1) the tenth of the corpus held out from training (called *Held Out*), and (2) an evaluation test set comprised of raw data collected from the entire ATIS corpus (called *ATIS Evaluation*).

After computing the perplexities, it was necessary to compare the values between the implemented models to determine if any one was significantly better than the others. After testing for normality across the different training samples using the Shapiro–Wilk test [4], we used the Student's t-test to test for significance. The remainder of this section outlines the procedure for performing these tests.

SHAPIRO–WILK TEST

The Shapiro–Wilk test is used to prove that a given statistical sample is taken from a population that has a normal distribution. The test is performed by calculating the W statistic, which "provide[s] an index or test statistic to evaluate the supposed normality of a complete sample." [4] The algorithm for computing W is described below.

Given a complete random sample of size n, $(x_1,x_2,...,x_n)$:

1. Order the observations to yield an ordered sample $y_1 \le y_2 \le \ldots \le y_n$.
2. Computef:

$$s^2 = \sum_{1}^{n} (y_i - \bar{y})^2 = \sum_{1}^{n} (x_i - \bar{x})^2$$

   where $\bar{x}$ is the mean of the random sample and $\bar{y}$ is the mean of the ordered sample.[1]

3. Compute the value for b:

$$b = \sum_{i=1}^{k} a_{n-i+1}(y_{n-i+1} - y_i)$$

   where $n = 2k$ when n is even and $n = 2k+1$ when n is odd, and the values for a_{n-i+1} are given in Table 5 of [4].

4. Now, calculate W:

$$W = \frac{b^2}{s^2}$$

1. One can easily prove that the two means are, in fact, equal.

and compare this computed value to the critical values for W in Table 6 of [4]. For this test, small values of W are significant, i.e., calculated values that are less than the critical values of the table indicate that the sample is not normally distributed.

Once we have used the Shapiro–Wilk test to establish the normality of the data sets, we can use the Student's t-test to compare them. This test is described in the next section.

STUDENT'S T-TEST

In Student's t-test, we directly compare the means and standard deviations of two normally-distributed samples. This is one of the preferred tests for a comparison of two cases. The test is done as follows:

1. First, we calculate $t_{observed}$ for our data using the following equation:

$$t_{observed} = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{(n_x - 1)s_x^2 + (n_y - 1)s_y^2}{n_x + n_y - 2}}\sqrt{\frac{1}{n_x} + \frac{1}{n_y}}}$$

   where $\bar{x}$ is the mean of the first case, $\bar{y}$ is the mean of the second case, n_x and n_y are the number of members in the first and second cases, respectively, and s_x and s_y are the standard deviations of the first and second cases, respectively.
2. After calculating $t_{observed}$, we need to obtain the critical value by looking up $t(n_x + n_y - 1)$ in a statistics reference (e.g., [1]). Generally, we will use t such that $p \le .05$.
3. Finally, we can use our observed and look-up values to determine significance. If $|t_{observed}| > t(n_x + n_y - 1)$, then we can reject the null hypothesis that the means for the populations are equal. This means that the difference between the sample means is significant.

Results

There were two data sets used to calculate the perplexities of the various models. The first data set, called *Held Out*, uses the portion of the data that was held out of training to test the model. The second data set, *ATIS Evaluation*, uses data from a set that was collected independently of the training data from the ATIS Evaluation Data corpus. Each data set was tested using both models and also using ten different samples of (90% of) the data from the training corpus. This section lists only the means and standard deviations across the ten samples for each model; the raw perplexities for each of the ten different samples are listed in the appendix.

TABLE 1. Mean Perplexities for all Models, Held Out Data (lowest perplexity listed in bold)

Model		N = 3	N = 4	N = 5	N = 6	N = 7
modkn–count		10.841947 (.218338)	9.779990 (.216310)	9.459193 (.209826)	9.347311 (.209268)	9.313084 (.212712)
modkn–extend/modkn–flex, threshold = 8		**10.772178 (.211728)**	**9.617073 (.208121)**	**9.233279 (.205649)**	**9.081552 (.201777)**	**9.019758 (.203821)**
modkn–diffd		12.627731 (.252498)	15.615920 (.402464)	20.711024 (.534459)	26.278446 (.584470)	31.084706 (.687919)
modkn–flex	threshold = 7	10.772178 (.211728)	9.617073 (.208121)	9.233279 (.205649)	9.081552 (.201777)	9.051555 (.204471)
	threshold = 6	10.772178 (.211728)	9.617073 (.208121)	9.233279 (.205649)	9.143066 (.203899)	9.112894 (.206526)
	threshold = 5	10.772178 (.211728)	9.617073 (.208121)	9.342093 (.207519)	9.250849 (.205857)	9.220288 (.208334)
	threshold = 4	10.772178 (.211728)	9.782679 (.213965)	9.502975 (.213736)	9.410167 (.212371)	9.379076 (.214649)
	threshold = 3	10.905230 (.220093)	9.903528 (.222181)	9.6203835 (.222377)	9.526429 (.220952)	9.494955 (.223217)
	threshold = 2	10.930852 (.219385)	9.926785 (.221156)	9.642978 (.221522)	9.548806 (.220254)	9.517254 (.222383)

TABLE 2. Mean Perplexities for all Models, ATIS Evaluation Data (lowest perplexity listed in bold)

Model	N = 3	N = 4	N = 5	N = 6	N = 7
modkn–count	13.799900 (.058049)	13.491072 (.077593)	13.492050 (.092120)	13.630996 (.095242)	13.723703 (.099762)
modkn–extend/modkn–flex, threshold = 8	13.646170 (.051014)	13.226394 (.062258)	13.224389 (.070841)	13.337332 (.071704)	13.415900 (.073314)
modkn–diffd	17.903963 (.114404)	26.956006 (.185196)	38.748564 (.284653)	50.880698 (.307067)	59.428086 (.260622)

TABLE 2. Mean Perplexities for all Models, ATIS Evaluation Data (lowest perplexity listed in bold)

Model		N = 3	N = 4	N = 5	N = 6	N = 7
modkn–flex	threshold = 7	13.646170 (.051014)	13.226394 (.062258)	13.224389 (.070841)	13.337332 (.071704)	13.372778 (.074446)
	threshold = 6	13.646170 (.051014)	13.226394 (.062258)	13.224389 (.070841)	13.284015 (.073091)	13.319321 (.075862)
	threshold = 5	13.646170 (.051014)	13.226394 (.062258)	13.202162 (.069345)	13.261688 (.071653)	13.296940 (.074467)
	threshold = 4	**13.646170 (.051014)**	**13.200967 (.063250)**	**13.176790 (.071971)**	**13.236204 (.074744)**	**13.271383 (.077611)**
	threshold = 3	13.706317 (.049769)	13.249434 (.061329)	13.225163 (.068994)	13.284793 (.071462)	13.320100 (.074289)
	threshold = 2	13.714211 (.050475)	13.257071 (.063374)	13.232787 (.071282)	13.292452 (.073652)	13.327779 (.076373)

The Shapiro–Wilk test [4] was run on selected data sets (modkn–count, modkn–extend, modkn–diffd, and modkn–flex [threshold = 4]) to establish the normality of the distributions across the ten training samples. All data sets proved to be normal (p[normality] > .95), indicating that the different models can be compared using a standard t-test.

Tables 1 and 2 clearly indicate that the selection of the discounting parameters using the number of extended contexts of an n-gram (rather than n-gram count) yields lower perplexity. Statistical tests show a significant difference in the ATIS evaluation data ($p < .05$) and in higher-order values in the held out data ($N > 4$, $p < .05$) between modkn–count and modkn–extend.We also find in the ATIS evaluation data significant differences between the modkn–extend perplexities and those for modkn–flex (threshold = 4), for $N > 5$ ($p < .05$). In addition, modkn–diffd yields significantly higher perplexities than any of the other modifications tested.

Conclusions

From the results listed above, a few facts about the different modifications to modified Kneser–Ney stand out. First of all, the significance tests prove that Chen and Goodman's choice of selecting the discounting parameter (in the lower order models) based on the number of extended contexts of an n-gram is clearly superior to selecting the discounting parameter based on the n-gram count. This seems sensible since the lower order equation uses the number of extended contexts in the calculation of the raw probability, rather than the n-gram count. In fact, the whole intent behind Kneser–Ney smoothing is to use a different distribution for lower order models that will add new information to the higher order model (by examining unique contexts rather than n-gram count), rather than simply duplicating the same information as can already be found in the higher order model. Therefore, by choosing a different discounting parameter, we are able to add new information to the lower order models.

The second fact that is made clear from the data is that the alternative method for calculating the values for the discounting parameters (modkn–diffd) does not improve the modified Kneser–Ney algorithm. In this case, the discounting parameters for the smoothing model were calculated using the frequency of extended contexts, that is, the number of n-grams that have one, two, three, or four extended contexts. The assumption was that, if the rest of the equation was based on the number of extended contexts rather than the n-gram count, then the discounting parameters should also be set according to the number of extended contexts. Further exploration of the reasoning given by Ries for the selection of the D parameters (outlined in Chen and Goodman) would be needed to attempt to explain the higher perplexities yielded by this modification.

Finally, the modkn–flex model yielded perplexities that were significantly lower than those for modkn–extend, but only for certain thresholds. In the Held Out data case, the perplexities are lowest for modkn–extend, so there is no advantage in leaving out singleton contexts for this data set. We can assume, however, that our actual test corpus for establishing baseline perplexities will be more of the nature of the ATIS evaluation data. In this case, the test data is more dissimilar to the training data than in the case of the Held Out data. For the ATIS evaluation data, we find lower perplexities for modkn–flex where the threshold is equal to four (compared to modkn–extend), but statistical tests find a significant difference only for higher order models ($N > 5$). Therefore, if the baseline measure we intend to use is trigrams, there is no advantage to using the (slightly) more complicated flex code.

All of the evidence presented suggests that the best implementation of the modified Kneser–Ney algorithm for our purposes, and given our test data, is modkn–extend, which is the model that was used by Chen and Goodman. If our aim was to establish a baseline for n-grams higher than three, the modkn–flex code might give better results. However, our intention is to use the modkn–extend algorithm to establish a baseline perplexity for trigrams, which we will then compare to perplexities based on language models that will be developed later.

References

[1] Beyer, W.H. *CRC Standard Mathematical Tables and Formulae, 29th edition.* Boca Raton: CRC Press, 1991.

[2] Chen, S.F., and J. Goodman. *An empirical study of smoothing techniques for language modeling*. Technical Report TR–10–98, Center for Research in Computing Technology (Harvard University), August 1998.

[3] Kneser, R. and H. Ney. Improved backing-off for m-gram language modeling. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, volume 1*. 1995. pp. 181–184.

[4] Shapiro, S.S., and M.B. Wilk. An analysis of variance test for normality (complete samples). *Biometrika,* 52:591–611 (1965).

Appendix: Raw Perplexities for each Left Out Section

HELD OUT DATA

The following tables list the raw perplexities for the held out test data, using each of the different variants of the modified Kneser–Ney algorithm.

TABLE 3. **Perplexities for modkn-count**

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
0	11.106794	10.049504	9.727847	9.610030	9.579785
1	10.803932	9.742997	9.416003	9.314682	9.282811
2	10.697874	9.691136	9.351543	9.226230	9.180228
3	10.880939	9.885522	9.582581	9.484371	9.451243
4	10.501997	9.445024	9.123965	9.021681	8.991502
5	11.265598	10.185952	9.833944	9.726277	9.698274
6	10.902483	9.750976	9.454313	9.327890	9.302928
7	10.818422	9.788891	9.460088	9.340334	9.311207
8	10.663621	9.579361	9.274683	9.170410	9.124429
9	10.777808	9.680534	9.366968	9.251203	9.208435
Mean	10.841947	9.779990	9.459193	9.347311	9.313084
Std. Deviation	.218338	.216310	.209826	.209268	.212712

TABLE 4. **Perplexities for modkn-extend**

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
0	11.029193	9.885600	9.511487	9.350395	9.292304
1	10.742017	9.575610	9.173364	9.029056	8.972115
2	10.634401	9.526619	9.114870	8.947236	8.872536
3	10.814173	9.717508	9.358965	9.218965	9.162931
4	10.433240	9.287181	8.911083	8.773192	8.712779
5	11.177359	9.996139	9.584883	9.427471	9.360249
6	10.830019	9.585201	9.223888	9.060586	9.003406
7	10.743895	9.638172	9.256238	9.105368	9.057961
8	10.600385	9.425405	9.045577	8.901856	8.831052
9	10.717101	9.533297	9.152440	9.001391	8.932248
Mean	10.772178	9.617073	9.233279	9.081552	9.019758
Std. Deviation	.211728	.208121	.205649	.201777	.203821

TABLE 5. Perplexities for modkn–diffd

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
0	12.983740	16.155488	21.504185	27.266888	32.242374
1	12.552227	15.594393	20.585566	26.215713	31.163737
2	12.428113	15.511806	20.511015	25.928479	30.649277
3	12.678865	15.954043	21.234222	26.689755	31.468312
4	12.217914	14.924814	19.774948	25.192693	29.803224
5	13.091066	16.212409	21.400565	26.927561	31.698201
6	12.613254	15.282159	20.377232	26.230413	31.310820
7	12.620707	15.700235	20.794587	26.374224	31.263022
8	12.509995	15.357738	20.363851	25.904358	30.465952
9	12.581428	15.466110	20.564065	26.054380	30.782144
Mean	12.627731	15.615920	20.711024	26.278446	31.084706
Std. Deviation	.252498	.402464	.534459	.584470	.687919

For the modkn–flex model, the data for any N below the threshold is the same value as was obtained using modkn–extend. Therefore, for the following tables, we list only the data points for N greater than or equal to the threshold.

TABLE 6. Perplexities for modkn–flex

Left Out Section	threshold = 7	threshold = 6		threshold = 5		
	N = 7	N = 6	N = 7	N = 5	N = 6	N = 7
0	9.325582	9.412976	9.387996	9.618157	9.518541	9.493281
1	9.001267	9.093166	9.065180	9.290574	9.209351	9.181008
2	8.912221	9.017113	8.981824	9.231621	9.132612	9.096872
3	9.193266	9.280540	9.254669	9.463107	9.383810	9.357651
4	8.747308	8.822230	8.796201	9.001370	8.911617	8.885325
5	9.400830	9.492571	9.465747	9.701756	9.608319	9.581167
6	9.025103	9.119204	9.083491	9.329191	9.223312	9.187192
7	9.083082	9.163718	9.141289	9.352994	9.259507	9.236844
8	8.864674	8.964528	8.927085	9.169330	9.087172	9.049216
9	8.962219	9.064911	9.025462	9.262834	9.174249	9.134324
Mean	9.051555	9.143066	9.112894	9.342093	9.250849	9.220288
Std. Deviation	.204471	.203899	.206526	.207519	.205857	.208334

TABLE 7. Perplexities for modkn–flex, threshold = 4

Left Out Section	N = 4	N = 5	N = 6	N = 7
0	10.047933	9.776099	9.674847	9.649172
1	9.772903	9.481994	9.399098	9.370171
2	9.673605	9.374056	9.273519	9.237227
3	9.891215	9.632267	9.551552	9.524925
4	9.421299	9.131361	9.040311	9.013639
5	10.166529	9.867129	9.772099	9.744484
6	9.755102	9.494554	9.386798	9.350038
7	9.785318	9.495786	9.400872	9.377863
8	9.595928	9.335221	9.251577	9.212934
9	9.716962	9.441288	9.350996	9.310302
Mean	9.782679	9.502975	9.410167	9.379076
Std. Deviation	.213965	.213736	.212371	.214649

TABLE 8. Perplexities for modkn–flex, threshold = 3

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
0	11.197374	10.201151	9.925172	9.822376	9.796310
1	10.879993	9.898431	9.603786	9.519825	9.490526
2	10.750135	9.778883	9.476073	9.374442	9.337756
3	10.951694	10.016999	9.754758	9.673017	9.646051
4	10.537343	9.515305	9.222474	9.130516	9.103578
5	11.297192	10.275526	9.972915	9.876867	9.848956
6	10.966301	9.877857	9.614031	9.504919	9.467696
7	10.883114	9.912116	9.618832	9.522689	9.499381
8	10.739409	9.721779	9.457652	9.372911	9.333762
9	10.849746	9.837228	9.558142	9.466732	9.425535
Mean	10.905230	9.903528	9.620384	9.526429	9.494955
Std. Deviation	.220093	.222181	.222377	.220952	.223217

TABLE 9. Perplexities for modkn–flex, threshold = 2

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
0	11.211236	10.213780	9.937459	9.834535	9.808437
1	10.912842	9.928316	9.632781	9.548567	9.519180
Mean	10.930852	9.926785	9.642978	9.548806	9.517254
Std. Deviation	.219385	.221156	.221522	.220254	.222383

TABLE 9. Perplexities for modkn–flex, threshold = 2

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
2	10.771412	9.798237	9.494829	9.392997	9.356237
3	10.979183	10.042142	9.779243	9.697297	9.670263
4	10.560966	9.536636	9.243149	9.150985	9.123986
5	11.327557	10.303144	9.999720	9.903413	9.875428
6	10.993505	9.902361	9.637881	9.528498	9.491183
7	10.890540	9.918880	9.625396	9.529186	9.505863
8	10.772998	9.752186	9.487233	9.402226	9.362955
9	10.888279	9.872165	9.592088	9.500354	9.459010
Mean	10.930852	9.926785	9.642978	9.548806	9.517254
Std. Deviation	.219385	.221156	.221522	.220254	.222383

ATIS EVALUATION DATA

The following tables list the raw perplexities for the ATIS evaluation test data, using each of the different variants of the modified Kneser–Ney algorithm.

TABLE 10. Perplexities for modkn-count

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
0	13.796250	13.468189	13.473419	13.620264	13.711583
1	13.736042	13.402007	13.392407	13.543655	13.646007
2	13.902056	13.662726	13.702912	13.854155	13.958922
3	13.728616	13.431838	13.431394	13.577018	13.669585
4	13.791340	13.473697	13.451825	13.600136	13.692278
5	13.759221	13.503607	13.520672	13.647075	13.747714
6	13.891268	13.559730	13.560544	13.709099	13.802564
7	13.808408	13.478628	13.443930	13.567799	13.652215
8	13.777521	13.409995	13.409712	13.534798	13.614543
9	13.808275	13.520301	13.533680	13.655959	13.741617
Mean	13.799900	13.491072	13.492050	13.630996	13.723703
Std. Deviation	.058049	.077593	.092120	.095242	.099762

TABLE 11. Perplexities for modkn-extend

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
0	13.670146	13.230009	13.231396	13.338832	13.419393
1	13.602959	13.156353	13.151225	13.271943	13.353193
Mean	13.646170	13.226394	13.224389	13.337332	13.415900
Std. Deviation	.051014	.062258	.070841	.071704	.073314

TABLE 11. Perplexities for modkn-extend

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
2	13.731964	13.344294	13.371031	13.489648	13.570779
3	13.587168	13.153276	13.144428	13.261603	13.339018
4	13.629052	13.205444	13.182802	13.295964	13.373741
5	13.617259	13.202289	13.211471	13.314997	13.397180
6	13.730559	13.282833	13.282616	13.398267	13.485364
7	13.677676	13.245849	13.229490	13.335099	13.406798
8	13.639578	13.166592	13.162665	13.274149	13.346226
9	13.675339	13.277001	13.276771	13.392817	13.467312
Mean	13.646170	13.226394	13.224389	13.337332	13.415900
Std. Deviation	.051014	.062258	.070841	.071704	.073314

TABLE 12. Perplexities for modkn–diffd

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
0	17.859619	26.873747	38.565671	50.499042	59.191492
1	17.824492	26.810235	38.669999	50.831801	59.478333
2	18.125620	27.236896	39.309692	51.406622	59.815774
3	17.777125	26.874868	38.730276	50.963085	59.495847
4	17.882762	26.860777	38.418617	50.548737	58.957140
5	17.782992	26.841783	38.595489	50.642877	59.274963
6	18.026111	26.960467	38.747332	51.069801	59.513553
7	17.957106	27.180039	38.917317	51.122396	59.586396
8	17.826217	26.712822	38.439228	50.582480	59.244791
9	17.977585	27.208424	39.092019	51.140136	59.722568
Mean	17.903963	26.956006	38.748564	50.880698	59.428086
Std. Deviation	.114404	.185196	.284653	.307067	.260622

For the modkn–flex model, the data for any N below the threshold is the same value as was obtained using modkn–extend. Therefore, for the following tables, we list only the data points for N greater than or equal to the threshold.

TABLE 13. Perplexities for modkn–flex

Left Out Section	threshold = 7	threshold = 6		threshold = 5		
	N = 7	N = 6	N = 7	N = 5	N = 6	N = 7
0	13.374957	13.290224	13.326217	13.205967	13.264681	13.300605
1	13.304717	13.214682	13.247314	13.134360	13.197735	13.230326
2	13.529012	13.437415	13.476627	13.343119	13.409364	13.448494
3	13.297928	13.205026	13.241195	13.129860	13.190391	13.226520
4	13.328979	13.234764	13.267627	13.155255	13.207108	13.239903
5	13.353645	13.263693	13.302192	13.189998	13.242135	13.280572
6	13.444003	13.346873	13.392434	13.259808	13.323955	13.369437
7	13.365294	13.283021	13.313099	13.209576	13.263028	13.293060
8	13.302637	13.224493	13.252874	13.134439	13.196134	13.224455
9	13.426613	13.339963	13.373627	13.259240	13.322349	13.355968
Mean	13.372778	13.284015	13.319321	13.202162	13.261688	13.296940
Std. Deviation	.074446	.073091	.075862	.069345	.071653	.074467

TABLE 14. Perplexities for modkn–flex, threshold = 4

Left Out Section	N = 4	N = 5	N = 6	N = 7
0	13.201606	13.177615	13.236204	13.272050
1	13.141124	13.119156	13.182458	13.215011
2	13.334543	13.333368	13.399565	13.438666
3	13.139968	13.116577	13.177047	13.213139
4	13.153373	13.103381	13.155030	13.187695
5	13.179778	13.167508	13.219555	13.257927
6	13.255170	13.232194	13.296207	13.341595
7	13.213532	13.177348	13.230668	13.260627
8	13.145357	13.113256	13.174852	13.203127
9	13.245215	13.227497	13.290455	13.323993
Mean	13.200967	13.176790	13.236204	13.271383
Std. Deviation	.063250	.071971	.074744	.077611

TABLE 15. Perplexities for modkn–flex, threshold = 3

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
0	13.712539	13.242546	13.218481	13.277251	13.313209
1	13.641299	13.178162	13.156132	13.219613	13.252257
2	13.761727	13.363444	13.362267	13.428607	13.467793
3	13.635027	13.186252	13.162779	13.223461	13.259681
4	13.694809	13.216834	13.166601	13.218499	13.251322
5	13.668499	13.229372	13.217056	13.269299	13.307815
6	13.784047	13.306806	13.283740	13.348003	13.393567
7	13.734700	13.268621	13.232286	13.285829	13.315912
8	13.689734	13.193696	13.161476	13.223299	13.251678
9	13.740793	13.308610	13.290807	13.354066	13.387765
Mean	13.706317	13.249434	13.225163	13.284793	13.320100
Std. Deviation	.049769	.061329	.068994	.071462	.074289

TABLE 16. Perplexities for modkn–flex, threshold = 2

Left Out Section	N = 3	N = 4	N = 5	N = 6	N = 7
0	13.709360	13.239476	13.215417	13.274174	13.310123
1	13.651026	13.187559	13.165513	13.229039	13.261707
2	13.765839	13.367437	13.366259	13.432619	13.471817
3	13.638506	13.189616	13.166137	13.226835	13.263064
4	13.698273	13.220177	13.169932	13.221843	13.254674
5	13.687549	13.247810	13.235477	13.287793	13.326363
6	13.788182	13.310799	13.287725	13.352007	13.397586
7	13.742923	13.276564	13.240207	13.293782	13.323884
8	13.693194	13.197030	13.164802	13.226640	13.255027
9	13.767257	13.334241	13.316404	13.379785	13.413549
Mean	13.714211	13.257071	13.232787	13.292452	13.327779
Std. Deviation	.050475	.063374	.071282	.073652	.076373