CALIBRATION FOR NATURAL LANGUAGE PROCESSING SYSTEMS

An Honor Thesis

Presented by

KHANH X. NGUYEN

Approved by:

Brendan O'Connor, School of Computer Science

Erik Learned-Miller, School of Computer Science

ABSTRACT

Title: **Calibration for natural language processing systems**

Author: **Khanh X. Nguyen**

Thesis/Project Type: **Project**

Approved By: **Brendan O'Connor, School of Computer Science**

Approved By: **Erik Learned-Miller, School of Computer Science**

Statistical natural language processing (NLP) models assign a posterior distribution to the set of possible outcomes and make their predictions based on those scores. Current performance metrics for NLP systems only take into account the model final decisions, which not only depends on the quality of the systems model but also the inference scheme, rather than explicitly reflect the quality of the prediction posterior distribution. In this thesis, I propose a metric for directly examining the quality of the posterior distribution, the calibration test. First of all, I will present the theoretical foundation for the concept of calibration. After that, I apply the test to numerous families of NLP models and show that the calibration is complementary to the traditional metrics in the sense that it provides a more comprehensive insights into the performances of NLP systems.

CHAPTER 1

INTRODUCTION

Researchers have long discovered that understanding natural language is much more difficult than simply putting the meanings of individual words together. Word meanings are affected by their surrounding contexts. The ambiguity of natural language motivated the use of the probabilistic model for tackling NLP tasks. A typical probabilistic model maps a linguistic structure to a posterior distribution over the space of all possible labels for the structure. The probability of an outcome can be interpreted as the degree of belief that the model holds for that possibility after learning the set of training instances. The model then make its decision based on an decision-making scheme.

Many NLP systems use MAP, i.e. choosing the most likely outcome, as their inference scheme. In this setting, the posterior distribution output by the model is neglected. Performance metrics such as accuracy score or F1-score measure how to the final decisions of the model align with the true labels. In complex systems that are structured as a cascade of multiple models, however, MAP inference seems to be inappropriate since the downstream models suffer from errors accumulated from upstream models. BLAH BLAH demonstrated that using the K-top prediction list yield better performance for BLAH BLAH. In a more advanced approach, Finkel et al. used MCMC to produce an approximate representation of the posterior distribution of the upstream models and passed it as input for the downstream models. Although the metrics of the final decisions of the pipeline reflects the quality of the entire system, there is a need for metrics to assess the posterior distribution of the intermediate models in order to obtain more insights into how they affect the overall performance.

Outside of the scope of NLP pipeline, knowledge about the posterior distribution of a model is also useful. Communicating the uncertainty of a model with its users is essential for calculating risk. This issue has been well-recognized in weather forecasting. Consider the task of predicting whether it will rain during a particular day or not. Suppose a weather forecasting model predicts that it will rain today with a belief score of 0.6. The delivered prediction from the model has the form There is X% chance that it will rain today. Using MAP inference with a threshold of 0.5 and reporting that It will rain today is a crude round-off and thus is more likely to lead to incorrect subsequent decisions in other tasks that rely on this piece of information.

Following BLAH BLAH, I propose a calibration-refinement framework for assessing the quality of posterior distributions of NLP models. Refinement measure is often encountered in the form of log loss or mean squared error. On the other, calibration measure receives less attention although it is complementary to refinement measure. The focus of this thesis is to develop a general procedure for measure calibration that is independent of the choice of model. The procedure takes a posterior distribution and the true data labels as input. The output can either be a visualizable calibration plot or a single calibration score depending on the need of the user.

In order to develop such procedure, first of all, a review the theoretical foundation of calibration is present. Several desirable characteristics of a well-calibrated model are shown. Next, a procedure for conducting calibration test on a model is described in details. Finally, the procedure will be applied to several families of NLP models. I show that BLAH BLAH BLAH.

CHAPTER 2

SUMMARY OF WORK OF PREVIOUS RESEARCHERS

The problem of miscalibration in prediction models that employ single-best inference scheme was addressed by Draper (1995). A Bayesian approach was proposed as an alternative. Finkel et. al (2009) apply this idea to tackle the problem of cascade NLP models. In this approach, the prediction posterior distribution of one task, which is approximated by a sample of the distribution, was passed as the input for other tasks. As suggested by the author, it is more general but easier to implement than approaches using K-best list (Sutton and McCallum (2005), Wellner et al. (2004), Huang and Chiang (2005), Toutanova et al. (2005)).

The concept of calibration was developed in the field of meteorology (Miller (1962), Murphy (1973)), referred to as validity or reliability. In Rubin (1982), it was argued that the applied statistician should Bayesian in principle and calibrated to the real world in practice. Murphy and Winkler (1982) proposed a general framework for forecast verification based on the joint distribution of forecasts and observations. They showed that the joint distribution of predictions and observations contains all of the information needed for assessing the forecast quality. They investigated it through two its Bayesian factorizations: the calibration-refinement factorization and the likelihood-base rate factorization. Their study is general in the sense that it can applied to any other type of prediction that produces a joint distribution between prediction labels and true labels. Rubin (2006) presented a method for validating software for Bayesian models using posterior quantiles. This idea will be applied in my thesis to formulate the notion of calibration for prediction problems where the predicted variable is continuous.

TODO: work on metrics for machine learning.

CHAPTER 3

EXPLANATION OF CURRENT METHODOLOGY AND GOALS

3.1 Calibration for binary probabilistic prediction tasks

Consider a binary probabilistic prediction problem, where each to-be-predicted instance i has a true label $y_i \in \{0, 1\}$. A stastical predictive model produces probabilistic predictions $q_i \in [0, 1]$ for each instance i, which represents a degree of belief that $y_i = 1$. For brevity, I will drop the subscripts and denote by y and q a generic true label and a generic probabilistic predition, respectively.

3.1.1 Calibration-refined framework

Given the joint probability $P(y, q)$, which contains all the information needed for analysing the quality of a set of probilistic preditions, Murphy and Winkler (1984) derived the calibration-refined framework from the following Bayesian factorization of $P(y, q)$:

$$P(y, q) = P(y \mid q)P(q) \tag{3.1}$$

Given a fixed value of q, the conditional probability $P(y = 1 \mid q^k)$ is called the *realistic frequency* with respect to the prediction q. It indicates how often the event $y = 1$ happens in reality among all instances whose predictions for that event are q. A model is said to be *perfectly calibrated* (or perfectly reliable) if its predictions match with their realistic frequencies. A formal definition for perfect calibration will be given in section BLAH BLAH. On the other hand, the marginal distribution $P(q)$ is an indicator for sharpness of the model. It specifies how often different values of predictions are assigned by the model. A model is said to be *refined* (or sharp) if $P(q)$ concentrates about 0 and 1. This charateristic of the model tells us that it is capable of discriminating between different types of data instances.

For a more concrete view of the calibration-sharpness framework, consider a standard probabilistic prediction problem: *weather forecasting*. When the forecaster give a prediction such as “There is 30% chance that it will rain tomorrow”, we should expect that among all the days that a weather forecaster predicts a 30% chance of rain, 30% will have rain. If the same expectation are met for all other predictions, we say that the forecaster is perfectly calibrated (or reliable). However, a forecaster who always predicts the climatological, i.e. the known long-term frequency of rain, will be perfectly calibrated although those predictions would be useless in discriminating between days with rain and days without rain. In this case, the weather forecasts are not *sharp* (or refined). Ideally, we would want forecasts that imply no uncertainty, either saying “100% chance of rain” or “0% chance of rain”.

As implied by equation (3.1), calibration and sharpness are orthogonal and complementary concepts. Perfect prediction implies perfect calibration and perfect sharpness. In the opposite direction, maintaining calibration allows posterior predictions to be more realistic whereas having sharpness in predictions reduces uncertainty in the decision-making process.

3.1.2 Perfect calibration

3.1.2.1 Definition

erfect calibration is defined as follows:

Definition 3.1. Let $p_q = P(y = 1 \mid q)$, the realistic frequency with respect to q. P binary predictive model is said to be *perfectly calibrated* if and only if:

$$p_q = q \qquad \forall q \in [0, 1].$$

As we can see, the left hand side of the equation represents a realistic frequency, which has to be equal to the corresponding probabilistic prediction on the right side for all possible values of q.

3.1.2.2 Adaptive bucketization

In practice, p_q is unknown so we have to estimate it using an *empirical frequency* $\hat{p}_q$ computed from a sample of data. However, there are usually not enough instances whose predictions is q to be used for computing $\hat{p}_q$ adequately. A solution for this problem is sort the predictions according to their values and partition them into groups of equal cardinalities. This technique is called "bucketization".

Choosing an appropriate bucket size is a non-trivial problem. As mentioned, if the size is to small the estimation for realistic frequencies will be too inaccurate to be useful. Conversely, as the bucket size grows larger, we are drifting away from verifying the true form of the condition in definition 3.1. An alternative approach for bucketization is to divide the interval $[0, 1]$ into equal-size sub-intervals. However, this approach does not guarantee the same degree of accuracy for all frequency estimates.

3.1.3 Miscalibration

3.1.3.1 Visualizing miscalibration

Definition 3.1 helps us identify perfectly calibrated models. But what about models that do not fall into that category? A quick way to imagine about miscalibration is to visualize it by a *reliability curve* or *empirical calibration curve.* After bucketizing the model predictions, we obtain a set of frequency-predition pairs (p_k, q_k). A calibration curve is a curve that smoothly connects the points (p_k, q_k) in a 2D-plot (Figure BLAH BLAH). We also want to show the perfect calibration curve, which coincides to the diagional line $x = y$, for comparison.

3.1.3.2 Measuring miscalibration

Having a metric to capture the miscalibration of a model is handy for model selection or tracking learning progress. Following DeGroot and Fienberg (1982), I will introduce concepts that are necessary for constructing such a metric.

Definition 3.2. Let p be real number in $[0, 1]$. A *strictly proper scoring rule* specified by an increasing function $g_1(x)$ and a descreasing function $g_2(x)$ is a function of x that has the following form:

$$f(x) = pg_1(x) + (1 - p)g_2(x) \tag{3.2}$$

and satisfies that $f(x)$ is maximized only at $x = p$.

Theorem 3.3. If $g_1(x)$ and $g_2(x)$ specify a strictly proper function rule, the overall score S for predictions for a probabilistic preditive model can be expressed in the form $S = S_1 + S_2$, where

$$\begin{aligned} S_1 &= E_q[p_q(g_1(q) - g_1(p_q)) + (1 - p_q)(g_2(q) - g_2(p_q))] \\ S_2 &= E_q[p_q g_1(p_q) + (1 - p_q)g_2(p_q)] \end{aligned} \tag{3.3}$$

It is not difficult to see that if the scoring rule is strictly proper then S_1 is maximized only when $p_q = q$ and S_2 is maximized when the values of p_q concentrate near 0 and 1. In fact, it can be proven that S_1 and S_2 have the following properties:

1. S_1 is zero only for perfectly calibrated model and negative otherwise.
2. If two model A and B are both perfectly calibrated and A is at least as sharp as B, the value of S_2 will be at least as large for A as it is for B.

Choosing $g_1(x) = (x - 1)^2$ and $g_2(x) = x^2$, then S_1 becomes the expected mean squared error between probabilistic predictions and the corresponding realistic frequencies:

$$MSE_{calib} = E_q[p_q - q]^2$$

3.2 Applications of calibration

3.2.1 Calibration for structure prediction

NLP researchers pay tremendous attention to linguistic structure prediction models (POS, NER, parsing). The caliration concept can also be applied to analyse the posterior predictions of these type of models. In this setting, y is not a single label but is a linguistic structure (parse (sub)trees, linear spans). A structure-predictive model assigns a probabilistic prediction q on each structure y.

We define $f(y)$ to be a binary-valued query function of the structure. For example, for a PCFG parsing model, $f(y)$ might denote whether particular span is an NP; for coreference resolution, it might denote whether the first and the sixth mentions belong to the same entity. We can the apply the same calibration framework for binary variables to assess calibration for the model.

Definition 4.1. Let $p_q = P(f(y) = 1 \mid q)$, the realistic frequency with respect to q. A structure-predictive model is said to be *perfectly calibrated* with respect to the query $f(y)$ if and only if:

$$p_q = q \qquad \forall q \in [0, 1].$$

Verification of calibration and measurement of miscalibration are conducted using the same methods described for binary varibables by regarding $f(y)$ as the binary variable.

3.2.2 Calibration for continuous variable

CHAPTER 4

REPORT AND DISCUSSION OF RESEARCH RESULTS

4.1 Comparison between miscalibrations of Hidden Markov Models and Condition Random Fields

We suspect that the characteristic of a probabilistic model is a factor that affects miscalibration in structure predictions. Therefore, we choose to compare between the two most widely used classes of models in structure predictions, Hidden Markov Models (HMMs) and Conditional Random Fields (CRFs). They are fundamentally different in their objective functions: HMMs are generative models which learn the joint distribution of the observations and the labels whereas CRFs are discriminative and model directly the conditional distribution of the observations given the labels. We perform our experiements on two common structure prediction tasks: part-of-speech tagging (POS) and named-entity recognition.

4.1.1 Part-of-speech tagging

4.1.1.1 Data

We extract articles from the Wall Street Journal (WSJ) from the CoNLL-2011 dataset for this experiment. The CoNLL has already been splitted the into training, development and testing sets so we only have to filter WSJ articles from those sets and join sentences in each set of articles into a single file. This process results in 11772 sentences for training, 1632 sentences for development and 1382 sentences for testing. The predictions we are testing is whether a word has the "NN" tag.

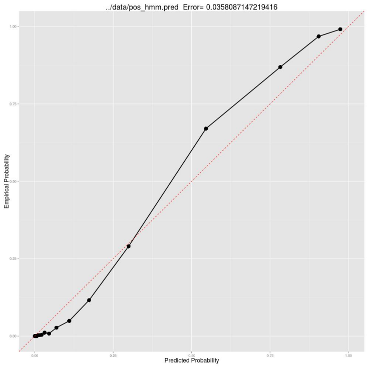

Figure 4.1. Calibration curve for HMM (POS), Acc = ??, CalibScore = ??

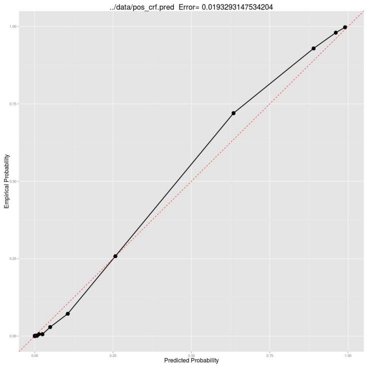

Figure 4.2. Calibration curve for CRF-Basic (POS), Acc = ??, CalibScore = ??

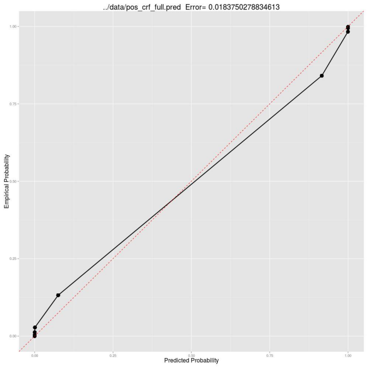

Figure 4.3. Calibration curve for CRF-Advanced (POS), Acc = ??, CalibScore = ??

4.1.1.2 Results

First of all, we compare a standard HMM with a CRF model with basic features (CRF-Basic). CRF-Basic contains only the emission features (pairs of the tag and the current token at each position) and the transition features (pairs of labels). Using CRF-Basic allows us to separate the characteristics of the model dependencies from the advantages of having features. The plots of calibration curves of the two models are shown in figure 4.1. CRF-basic intuitively produces a better curve than HMM. Concretely, the miscalibration of CRF-Basic is roughly 1.8 times larger than that of HMM (0.019 vs. 0.035). Moreover, as seen from the distributions of points in the plots, CRF-Basic produces sharper predictions.

To measure fully the power of the CRF model, we add more features to it, including surrounding words, word shape, word length, prefixes and suffixes. This model, called CRF-Advanced, achieves a 96% accuraccy on the task. It produces an extremely good posterior distribution. Figure 4.3 shows its perfect calibration curve, which is just slightly off the perfect calibration line (calibration score = 0.018).

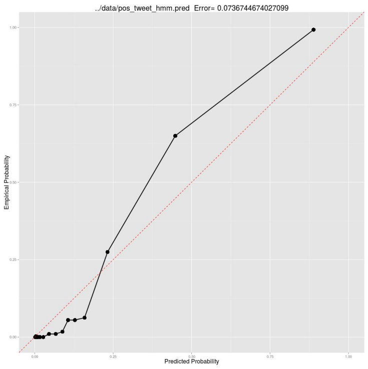

Figure 4.4. Calibration curve for HMM (POS Tweet), Acc = ??, CalibScore = ??

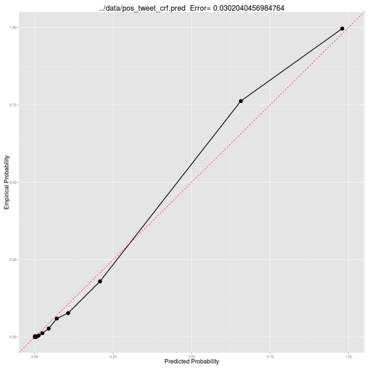

Figure 4.5. Calibration curve for CRF-Basic (POS Tweet), Acc = ??, CalibScore = ??

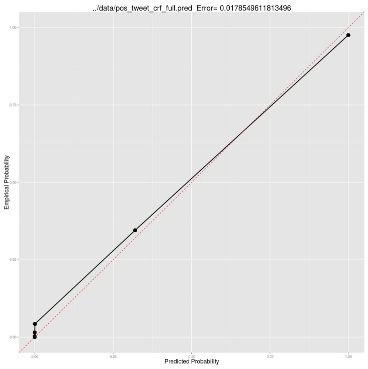

Figure 4.6. Calibration curve for CRF-Advanced (POS Tweet), Acc = ??, CalibScore = ??

4.1.2 Twitter part-of-speech tagging

4.1.2.1 Data

We repeat our comparision between HMMs and CRFs on a harder task, predicting POS tags for tweets. We use the ARK's Twitter POS data set (CITE NOAH), which consists of 1000 sentences for training, 327 sentences for development, 500 sentences for testing. The predictions we test is to predict whether a word has the "V" tag.

4.1.2.2 Data

We conduct the same experiments as in Section BLAH BLAH and obtain similar patterns. CRF-Basic's miscalibration is about half HMM's (Figure 4.4). On the other hand, equipped with better features, CRF-Advanced demonstrates a significant improvement from CRF-Basic, reducing further the miscalibration level by one half. It should also be noticed that CRF-Advanced does not give perfectly accurate predictions (87% accuracy) but those are reliable predictions.

4.2 Calibration analysis on synthesis data

4.2.1 Data

Approximating calibration statistics is very difficult since the distribution of the true labels is unknown. Therefore, we investigate the behavior of calibration statistics on synthesized set of prediction-observation pairs that mimics common NLP data distributions.

Each prediction-observation pair is generated as follows. First of all, the value of the prediction is drawn from a beta distribution. Then, the observation is obtained by sampling from a Bernoulli distribution whose parameter is a transformation of the value of the prediction. To obtain a perfectlycalibrated set of pairs, we use the identity transformation. For uncalibrated condition, we use this function $t(p)$:

$$t(p) = \begin{cases} \max(0, p - k), & \text{if } 0 \leq x \leq 0.5 \\ \min(0, p + k), & \text{otherwise} \end{cases}$$

where k ∈ [0, 0.5].

4.2.2 Effect of bin size on calibration score

We investigate the the effect of varying the bin size on the value of the MSE calibration score. Theoretically, as we double the bin size, the score will not increase. This fact is obtain by using Jensen's inequality, leveraging the fact that the quaratic function is convex. In our experiment, we vary the bin size from 2^1 to 2^16 to calculate the MSE calibration score on a data set consists of 10^5 pairs. Our result (Figure BLAH BLAH) supports the theoretical hypothesis. The score monotonically decreases as the bin size exponentally increases. We also alter the parameters of our beta distribution and witness the same pattern. We attempt to generalize this pattern to a contious range of bin size values. Figure BLAH BLAH portrays the behavior of the score of a perfectly calibrated predictor as the bin size goes from 10^3 to 5.10^4 with a step size of 10^3. As we can see, although the points do not always monotonically decrease, we see a similiar trend as the log-scale plot. However, for an uncalibrated predictor, the score is much more unpredictable (Figure BLAH BLAH).

4.2.3 Effect of sample size on calibration score

As pointed out by Foster (1998), we expect the calibration score of a perfectly calibrated predictor to go to zero as the sample size goes to infinity. We set up experiment to verify this fact. Using a range of sample size from 10^4 to 5.10^4, we compute the calibration score for three set of predictions: perfectly calibrated (PERFECT), uncalibrated using the function t(p) as true distribution with $k = 0.1$ (UNCALIB). For each experiement, we set the bin size to be the square root of the sample size. We observed distinguishing pattern between PERFECT and UNCALIB. The points in the CALIB's plot clearly approach zero while those of the UNCALIB's plot converge weakly.

CHAPTER 5

A INTRODUCTION TO SHEEP

Is there life afters:m:w sheep? [1] Yes, I say there is.

5.1 Pulling the wool over your eyes

Sheep are fabulou creatures. The noises they make are truly stupendous [2]. We also want to refer to figure 5.1 here. Here' some verbatim text to screw us up:

```
xxx := y;
xy := x;
```

5.1.1 All about sheep noises

Lots of text here just to fill up some space so we can be sure that we really are double-spacing and doing all the other things that might be necessary in formatting a dissertation to U.Mass. guidelines. We're also going to have another figure here, figure 5.2, just for fun, and to make sure that the list of figures is formatted correctly. Now it's time for table 5.1. We really are going to need a third figure, figure 5.3, two more tables, table 5.2 and table 5.3 and a fourth figure, figure 5.4, just to really make sure.

Table 5.1. Some numbers.

Type of Animal	Minimum Observed	Average Observed	Maximum Observed
Cats	12	20	24
Dogs	20	20	20

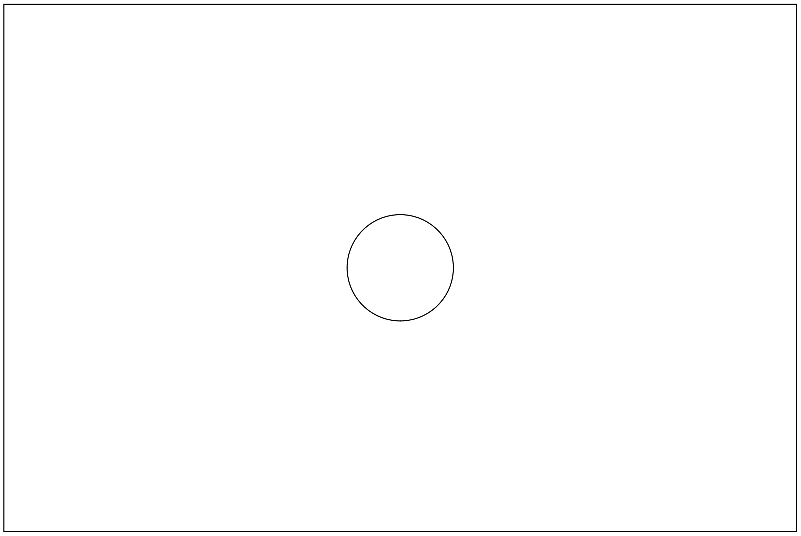

Figure 5.1. A circle in a square.

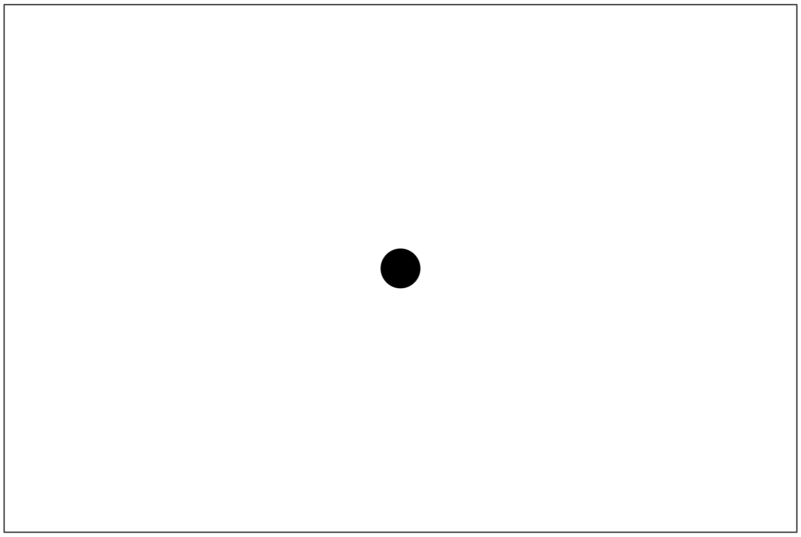

Figure 5.2. A disc in a square.

Table 5.2. More numbers.

Type of Animal	Arms	Legs	Ears
Person	2	2	2
Dog	0	4	2

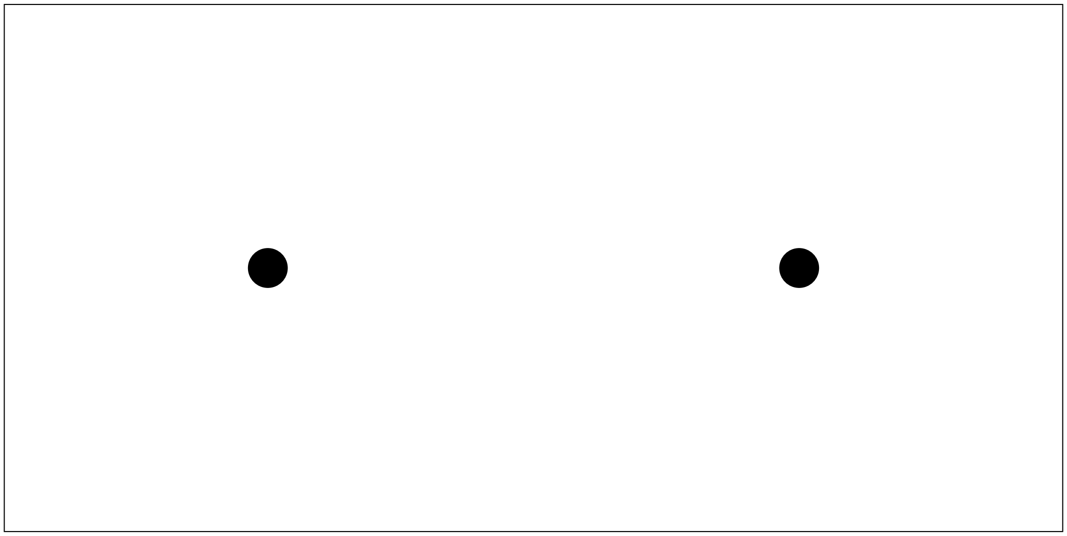

Figure 5.3. Two discs in a rectangle.

Table 5.3. Even more numbers; together with a caption long enough to ensure that multi-line caption formatting works correctly. If you want a shorter caption to appear in the Table of Figures you're going to have to put the shorter caption in the `[]` as shown in this example.

x	1	1	1
y	2	2	2
z	3	3	3

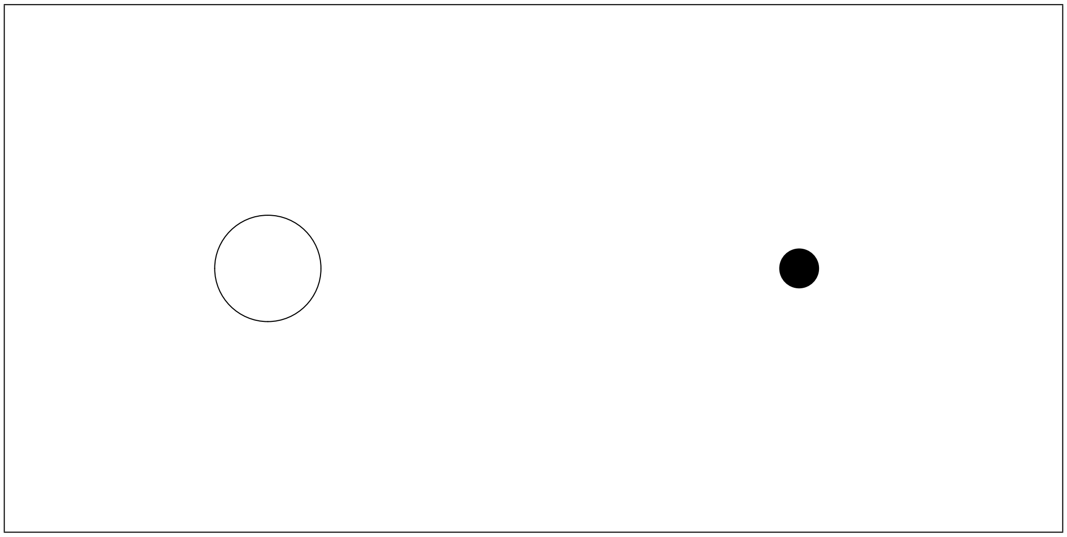

Figure 5.4. A circle and a disc in a square. We want this caption to be very long to ensure that the formatting of very long captions is handled correctly. The case of short captions has already been dealt with.

5.1.1.1 Baahs

5.1.2 Even more about sheep noises

5.1.3 And yet more about sheep noises

5.2 What about wolves?

What about wolves?[1]

5.3 What about shepherds?

What about shepherds? I don't really know, but I want some text here to fill things in so that I can verify that everything is OK.[2]

[1] To be fair, some wolves are probably nice...

[2] Some shepherds are good, some are bad. The reader is referred to Mary and The Boy Who Cried Wolf for further insight into this much-debated issue. (This needs to be a very long footnote so we can test the spacing between lines on a footnote.)

5.3.1 A subsection

This is a subsection of the subsection about shepherds.

5.3.2 Another subsection

This is another subsection of that section.

5.3.2.1 A subsubsection

This is a subsubsection of that subsection that will in turn havae a paragraph with a pair of subparagraphs. I am aware that I shouldn't have only one subsubsection in the subsection...

5.3.2.1.1 A Paragraph This is the text associated with this paragraph. I really want enough text to make it look like a paragraph. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah.

5.3.2.1.1.1 A Subparagraph This is the text associated with this subparagraph. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah.

5.3.2.1.1.2 Another Subparagraph Better not have subparagraphs without text in them. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah.

5.3.2.1.2 Another Paragraph Baah, baah, baah.

Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah.

5.3.2.2 Another Subsubsection

With some text. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah. Baah, baah, baah.

CHAPTER 6

SHEEP AND GRASS

6.1 Introduction

Grass is a wonderful food... Baah, baah, baah.

CHAPTER 7

A WONDERFULLY LONG CHAPTER TITLE THAT IS THIS LONG IN ORDER TO TEST THE CHAPTER HEADING STUFF

Note that we shouldn't really have a chapter heading with no body, so here is a body for this chapter. Baah, baah, baah.

7.1 The antidisestablishmentarainism supercalifragilisticexpialidocious longlonglonglonglongword

A `quotation`:

> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut nibh orci, molestie non vehicula ac, ultricies quis purus. Nunc euismod metus vel nulla sodales quis tempus nisi varius. Sed ornare pulvinar bibendum. Ut egestas mollis nisi vel cursus.

. . . and a `quote`:

> Ut dolor libero, blandit tristique accumsan non, viverra a magna. Sed pretium sollicitudin neque, sit amet ornare lorem convallis ac. Fusce mollis gravida aliquam. Nullam vulputate turpis vitae orci porttitor auctor. Donec in auctor erat.

APPENDIX A

THE FIRST APPENDIX TITLE

...

APPENDIX B

THE SECOND APPENDIX TITLE

...

BIBLIOGRAPHY

[1] Barrett, Daniel J., Ridgway, John V. E., and Wileden, Jack C. Why there are no sheep in our work. In *Proceedings of the Third Sheep Conference* (Edinburgh, Scotland, Jan. 1997), Ian McPherson Sheepish, Ed., American Sheepherders Society, Sheepdip and Associates, pp. 39–45.

[2] Scrooge, Ebenezer, and Shepherd, Alan. On the growth of green in space. *Journal of Astrophysical Economics 3*, 4 (August 1992), 47–89.