2

Relevance

2.1 The many faces of relevance

The notion of relevance serves as the foundation for the field of Information Retrieval. After all, the purpose of retrieval systems is to retrieve relevant items in response to user requests. Naturally, most users have a fairly good idea of what relevance is – it is a representation of their *information need*, a reflection of *what they are searching for*. However, in order to build and test effective retrieval systems we must translate the intuitive notion of relevance into a strict formalism, and that turns out to be somewhat of a challenge.

2.1.1 A simple definition of relevance

One of the simplest, and also most widely used, definitions of relevance is that of a binary relation between a given information item (document D) and the user's request (query Q). We might assume that the document is represented by a set of key words, appropriately reflecting its contents. Similarly, the user's request Q is a set of key words that represent the user's interest. Given these representations, we may say that a relevance relation between D and Q holds if there is a substantial overlap in meaning between the keyword sets of D and Q. Under this view, relevance of D does not depend on any factors other than representations of D and Q. Specifically, it does not depend on the user who issued the request, the task that prompted the request, or on user's preferences and prior knowledge. Similarly, in this simple definition relevance does not depend on any other documents D' in the collection, whether or not they have been examined by the user, or even judged relevant or non-relevant. It also does not depend on any other requests Q' to which D was previously judged relevant or non-relevant.

When relevance is defined as above, it is often called *system-oriented* or *algorithmic* relevance. The definition has been challenged and deemed inadequate on numerous occasions. In his comprehensive review of various formulations of relevance Mizarro [91] cites 160 papers attempting to define various

aspects of relevance. In this section we will briefly highlight a few of the most popular arguments about the nature of relevance. A reader yearning for a more complete exposition of the subject is invited to examine the work of Saracevic [120], Robertson [113], Harter [53] and Mizarro [91].

2.1.2 User-oriented views of relevance

Discussions of the proper definition of relevance began in 1959 when Vickery [142, 143] argued for a distinction between "relevance to a subject" and "user relevance". The former refers to a degree of semantic correspondence between the user's request and an item returned by the system in response to that request. The latter is a reflection of how much the user likes the retrieved item, taking into account his task and previously seen items. The distinction between the two can be explained if we imagine that our collection contains two near duplicates of the same item. If one of them is semantically relevant to the request, so will be the other. However, the user may not be satisfied with the second item, since it is completely redundant.

A somewhat different dichotomy of relevance was mentioned by Maron and Kuhns, who incidentally were the first to treat relevance probabilistically. In [85] they consider the distinction between the user's request and the underlying *information need*. The request is a surface representation of information need, it is observable and readily available to the system and to other users. The information need itself is an abstract concept that only the user himself is aware of. A document that appears relevant to the request may be completely irrelevant to the underlying information need. This happens frequently because of ambiguity inherent in human communication; the problem is further aggravated by the users' tendency to keep their requests short and devoid of little redundancies that would be so helpful to a search engine.

Belkin et al. [9, 10] make a further abstraction of a user's *information need*. They introduce a concept of ASK (*Anomalous State of Knowledge*), signifying the fact that the user himself may not be fully aware of what he is searching for. The concept of information need is completely abstract, it is not observable; the user has only a perception of that need, and that perception can change during the course of a searching session. The same idea is raised in a number of other publications. For example, Ingwersen [59], coins the acronyms ISK and USK, referring respectively to *Incomplete* and *Uncertain* States of Knowledge.

Foskett [46, 47] and later Lancaster [73] make an interesting argument that the nature of relevance to a large degree depends on the person who is making a judgment. In their definition, the term *relevance* refers to a "public" or "social" notion, where the judgment is made by an external expert or collective, and not by the user who posed the request in the first place. For the case when the judgment is made by the user himself, they coin the term *pertinence*. In light of previous discussion by Maron and Kuhns [85], relevance

is a relation between a document and the request, while pertinence is a relation between a document and the underlying information need.

In opposition to most of the distinctions drawn above, Fairthorne [43] makes a sobering argument for a strict definition of relevance, involving only the words contained in the document and the query. Otherwise, he claims, for any given document and any request, no matter how distant, we could hypothesize a situation where that document would in fact be relevant to the request.

2.1.3 Logical views of relevance

A number of authors attempted to define relevance formally, through logical constructs on the semantics of the documents and requests. One of the first formal attempts is due to Cooper [25], who defines relevance in terms of entailment (as in theorem-proving). Suppose q represents a logical proposition corresponding to a user's request. Let s be a proposition reflecting the meaning of some given sentence. Cooper says that s is relevant to q if s is a necessary assumption one needs to make in order to prove q. Formally, it means that s belongs to a minimal set S of propositions that entail q:

$$rel(s,q) \iff \exists S : s \in S, S \models q, S - s \not\models q$$

A document D is considered relevant to the request if it contains at least one sentence s that is relevant to q. Cooper's definition is attractive in that it allows relevance to be defined on a sub-document level. This makes it reasonable to allow that documents may discuss different topics and that a single document may be relevant to very different requests. The definition also allows relevance to be extended into *novel* or *marginal* relevance, which will be discussed later in this section.

There are two main criticisms of Cooper's definition of relevance. One is operational, and has to do with the inherent difficulty of transforming natural language into logical propositions. The other is conceptual – Cooper's relevance is a binary relation between a query (expressed in logical form) and a document. The user is not in the picture, and neither is there any way to factor in the task that prompted the search in the first place. Consequently, with Cooper's definition a document ought to be judged relevant by every user that happens to generate q as their request, regardless of the task they are faced with or their personal preferences. An attempt to improve Cooper's definition was made by Wilson [150], who introduced the idea of *situational relevance*. Wilson's concept of relevance includes the situation in which the search is performed, user's goals, as well as the information already known to the user prior to examining a given document.

Cooper's definition of relevance experienced a strong revival when it was re-formulated in 1986 by Van Rijsbergen [138]. Van Rijsbergen's work was followed by a large number of publications describing relevance in terms of

various formal logics [98, 99, 100, 101, 87, 122, 31, 32]. One important characteristic of most, if not all of these re-formulations is that they try to replace strict logical entailment ($d \models q$) with some weaker, but more tractable form. For example Van Rijsbergen [138, 139, 140] replaces the strict entailment with *logical implication* (denoted $d \rightarrow q$). Bruza [19, 20, 18] introduces a concept of *plausible entailment* and argues that a document d should be considered relevant as long as the query is at least plausibly entailed by some part of d. Lalmas and Van Rijsbergen [70, 71, 72] use modal logic and situation theory, treating a document as a *situation* and declaring it relevant if a flow of information from a document may lead to another situation, which in turn could strictly entail the query.

2.1.4 The binary nature of relevance

In most definitions relevance takes a form of a binary relation between a document and a query – a document is either relevant or it is not. A number of authors attempted to introduce graded notions of relevance, either in terms of discrete categories [38, 39, 65], or in terms of points or confidence intervals on the real line [42, 60]. While interesting in their own right, these definitions usually run into a number of practical difficulties. It turns out that graded relevance judgments are more costly to obtain, and the agreement between different judges is lower than for binary relevance. In addition, there is some evidence that human judges naturally prefer to use the end points of a given scale, hinting at the dichotomous nature of relevance [61]. Also, the methodology for evaluating retrieval effectiveness with graded relevance is not nearly as developed as it is for the case of binary relevance, where there is a small set of universally accepted performance measures.

Decision-theoretic view of relevance

When relevance is represented with real numbers, it is natural to start treating it as a measure of *utility* to a particular task, as was done by Cooper [28] and others. Zhai [157] has taken this view a step further, re-casting the entire retrieval process as that of minimizing the *risk* associated with missing usefull documents or presenting the user with garbage. In a noteworthy departure from related publications, Zhai's framework also incorporates a notion of a presentation strategy, where the same set of retrieved documents may result in different risk values, depending on how the documents are presented to the user.

2.1.5 Dependent and independent relevance

Most operational systems assume that relevance of a given document is independent of any other document already examined by the user, or of any

other unexamined document in the collection. The assumption is motivated by a number of practical concerns. First, non-independent relevance judgments are considerably more expensive to obtain, since the judgment for a particular document will depend on the order in which all documents are presented to the judge. Second, retrieval algorithms themselves become computationally expensive when we have to search over the subsets of a large collection of documents. However, assuming independent relevance often takes us too far from the reality of an information seeking process, so a number of alternative definitions exist and will be outlined below.

Relevance and novelty

Redundancy of information is perhaps the most common reason for considering alternative, non-independent definitions of relevance. If a collection contains two near-duplicate documents the user is unlikely to be interested in reading both of them. They both may be topically relevant to the request, but once one of them is discovered, the second one may become entirely redundant, and irrelevant for information seeking purposes. To reflect the value of novelty, Carbonell and Goldstein [21] proposed the concept of *maximal marginal relevance* (MMR). Their idea was to provide a balance between the topical relevance of a document to the user's request, and redundancy of that document with respect to all documents already examined by the user. Allan and colleagues [6, 4] recognized the fact that novelty and redundancy must also be addressed on a sub-document level. For example, a document may be mostly redundant, but may contain a small amount of novel and very pertinent information. Something like this often happens in news reporting, where journalists tend to re-use a lot of previously reported information, interspersing it with important new developments. In Allan's formulation there are two separate definitions of relevance – topical relevance and novel relevance, and system performance is evaluated independently with respect to both of them. The importance of novelty in ranking has led to the establishment of the *Novelty Track* within the Text Retrieval Conference (TREC) [49], which has attracted a large number of publications in the last few years.

Relevance of a set of documents

If the information need of the user is sufficiently complex, it may be possible that no individual document completely satisfies that need by itself. However, information pooled from several documents may be sufficient. In this case the assumption of independent relevance clearly does not hold and we have to conceptualize relevance of a *set of documents* to the information need. For a single document, we may define a notion of *conditional* relevance, where conditioning is with respect to other documents included in the retrieved set. One of the first formal models of conditional relevance is due to Goffman [48], who defines the relevant set as a *communication chain* – a closed sequence of

documents where relevance of a given document is conditioned on the previous document in the sequence (and must exceed a certain threshold for the document to be included). Goffman's model was evaluated by Croft and Van Rijsbergen [36] with results suggesting that the model, while considerably more expensive from a computational standpoint, was not superior to simpler forms of retrieval.

A major step away from independent document relevance was taken by Van Rijsbergen when he defined the now famous *cluster hypothesis*[137]:

> "Documents that cluster together tend to be relevant to the same requests."

The hypothesis was tested in a large number of studies, notably [62, 141, 144, 55, 82]. However, with few exceptions all these studies evaluate relevance at the level of individual documents in a cluster, rather than relevance of the cluster as a whole. Clustering was used primarily as a different way to organize retrieved documents.

Aspect relevance

In some cases, a complex information need can be broken down into smaller independent components. These components are often called *aspects*, and the goal of the retrieval system is to produce a set of documents that cover as many aspects of the overall need as possible. In this setting, it is common to introduce *aspect relevance*, which is topical relevance of a single document to a particular aspect of the overall need. Similarly, *aspect coverage* refers to the number of aspects for which relevant documents exist in the retrieved set. Aspect relevance and aspect coverage have been extensively studied in the Interactive Track of the TREC conference. A formal mechanism for modeling aspect utility was integrated by Zhai [157] into his risk-minimization framework.

2.2 Attempts to Construct a Unified Definition of Relevance

Faced with a growing number of definitions of relevance, several authors attempted to come up with a unified definition, which would classify and relate various notions of relevance. Some of the more prominent attempts were made by Saracevic [121], Mizarro [91, 92] and Ingwersen [30]. In this section we will briefly describe one particularly interesting proposal to formalize relevance by embedding it in a partially ordered four-dimensional space.

According to Mizarro [92], almost any reasonable definition of relevance can be represented as a vector consisting of four variables: *Information*, *Request*, *Time*, *Components*. The four dimensions of Mizarro's space have the following interpretations:

1. **Information type.** The first dimension of relevance is the kind of information resource for which we are defining relevance. In Mizarro's definition, this dimension can take one of three values: *document*, *surrogate* and *information*. Here *document* refers to the physical item a user will receive as the result of searching – the full text of a document, or, in the case of multi-media retrieval, a complete image, a full audio or video, file. *Surrogate* refers to a condensed representation of an information item, such as a list of keywords, an abstract, a title, or a caption. *Information* refers to changes in the user's state of knowledge as a result of reading or otherwise consuming the contents of a document. Note that information is a rather abstract concept, it depends on user's state of knowledge, his attentiveness, his capacity to comprehend the contents of the document and an array of other factors.
2. **Request type.** The second dimension of relevance specifies a level at which we are dealing with the user's problem. Mizarro defines four possible levels: *RIN*, *PIN*, *request* and *query*. The first (*RIN*) stands for Real Information Need and defines the information that will truly help the user solve the problem that prompted him to carry out a search in the first place. Needless to say, the user may not even be fully aware of what constitutes his real information need, instead he *perceives* it, and forms a mental image of it. That image is called *PIN*, or Perceived Information Need. Once the user knows (or rather thinks that he knows) what he is searching for, he formulates a *request*. A request is a natural language specification of what the user wants to find, something that might be given to a knowledgeable librarian or an expert in the field. A request is a way of communicating the *PIN* to another human being. Finally, this request has to be turned into a *query*, which is something that can be recognized by a search engine, perhaps a list of key words, a boolean expression or an SQL query. A query is a way of communicating the request to a machine.
3. **Time.** The third dimension of relevance reflects the fact that searching is not a one-shot process. The user may not see any relevant items in the initial retrieved set, and this may prompt him to re-formulate the *query*, perhaps changing the boolean structure, or adding some additional keywords. If the user does find relevant items, information in these items may prompt him to formulate different *requests*, or perhaps even force him to re-think what it is he wants to find, thus changing the perception of his information need (*PIN*). The real information need (*RIN*) stays constant and unchanging, since it is refers to what the user will ultimately be satisfied with. Mizarro endows the third dimension of relevance with a discrete progression of time points: $\{i_0, p_0, r_0, q_0, q_1, \ldots, r_1, q_{k+1}, \ldots, p_1, q_{m+1}, q_{n+1}, \ldots\}$. Here i_0 refers to the time the real information need (*RIN*) came to existence, p_0 is the time when the user perceived it, and decided what he wants to search for, r_0 is the time when he formulated a natural-language request, and q_0 is the time when that request turned into a query for a search engine. Proceeding further, q_1 is the time of the first re-formulation

of the request into a different query, r_1 is the first re-formulation of the natural-language request, and p_1 is the first time when the user changed the perception of what he wants to find. A request change r_i is always followed by one or more attempts to re-formulate the query q_j, and for every change in user's *PIN* there is at least one attempt to formulate a new request.

4. **Components.** The final dimension of relevance in Mizarro's framework specifies the nature of relationship between the first and the second dimension. It can be *topical* relevance, *task* relevance, *context* relevance, or any combination of the three. *Topical* relevance is concerned with semantic similarity in the content of the two items. *Task* relevance specifies that the item or information contained in it is useful for the task the user is performing. *Context* includes anything that is not covered by the topic and the task. Mizarro's *context* is a kind of "miscellaneous" category that subsumes the notions of novelty, comprehensibility, search cost, and everything else that does not seem to fit elsewhere in his formulation.

Mizarro [92] argues that his framework provides a useful tool for classifying and comparing various definitions of relevance. For example, the simple definition we provided in section 2.1.1 corresponds to *topical* relevance of a *surrogate* to the *query* at time q_0. To contrast that, Vickery's [142, 143] notion of "user relevance" relates the *information* in a document to the real information need (*RIN*) with respect to *topic*, *task* and *context* together.

At this point it is helpful to pause and admit that most practical retrieval models assume a fairly conservative definition of relevance restricted to a small portion of Mizarro's space. Most operational systems are concerned exclusively with *topical* relevance of full-text *documents* to natural-language *requests*. While there is active research on modeling *task* relevance, or taking *context* into account, the majority of experimental studies address only topicality. Any type of relevance with respect to perceived information need (*PIN*) is difficult to assess since there is only one user that knows what that *PIN* is. Furthermore, if we are dealing with *PIN*, we cannot cross-validate relevance judgments across different annotators. Modeling the real information need (*RIN*) is next to infeasible, since no one (not even the user) really knows what that is. Similarly, with very few exceptions (e.g. [81]) no one attempts to explicitly model the *information* contained in a given document – we simply do not have the frameworks that are simultaneously rich enough to represent arbitrary human discourse and robust enough to work on real-world data. On the other end of the spectrum, document *surrogates* and boolean *queries*, while quite popular in the past, are rarely used by the modern retrieval engines. Most current systems use full-text indexing, often with positional information and directly support natural-language requests.

2.2.1 Relevance in this book

For the scope of this book, we will concern ourselves with the popular view of relevance. In terms of Mizarro's classification, we will be constructing a formal model of relevance dealing with:

1. **Documents.** We will be operating on complete surface representations of information items (i.e. full text of documents, image bitmaps, segments of video, etc.)
2. **Requests.** We are concerned only with observable natural-language requests, for which we can obtain relevance judgments. However, our model will involve a notion similar to the real information need (**RIN**), which will play the role of a latent variable. We will use the words "query" and "request" interchangeably.
3. **Non-interactive.** We will not be modeling any evolution of user's information need. Our model explicitly accounts for the fact that a single information need can be expressed in multiple forms, but we do not view these in the context of an interactive search session.
4. **Topicality.** We will be interested exclusively in *topical* relevance, i.e. the semantic correspondence between a given request and a given document. We will not be addressing issues of presentation, novelty, or suitability to a particular task.

2.3 Existing Models of Relevance

This book is certainly not the first endeavor to treat relevance in probabilistic terms. Some of the more prominent examples are the 2-Poisson indexing model developed independently by Bookstein and Swanson [15, 16] and Harter [52], the probabilistic retrieval model of Robertson and Sparck Jones [117], the probabilistic flavors [123] of Van Rijsbergen's logical model [139], the inference-network model developed by Turtle and Croft [135, 134], the language modeling approach pioneered by Ponte and Croft [106, 105] and further developed by others [90, 56], and the recent risk minimization framework of Zhai and Lafferty [68, 157]. While we cannot provide a comprehensive review of all probabilistic models, we will devote the remainder of this chapter to a brief description of those models that had a particularly strong influence on the development of our generative model.

2.3.1 The Probability Ranking Principle

At the foundation of almost every probabilistic model of IR lies an intuitive principle that a good search system should present the documents in order of their probability of being relevant to the user's request. It appears that this idea was first formally stated by Cooper in a private communication to Robertson, who published it in the following form [114]:

> The Probability Ranking Principle (PRP): If a reference retrieval system's response to each request is a ranking of the documents in the collections in order of decreasing probability of usefulness to the user who submitted the request, where the probabilities are estimated as accurately as possible on the basis of whatever data has been made available to the system for this purpose, then the overall effectiveness of the system to its users will be the best that is obtainable on the basis of that data.

Symbolically, if D is used to denote every observable property of a given document in a search task, and if R is a binary variable indicating whether or not that document is relevant to the search request, then optimal performance would result from ranking documents in the decreasing order of $P(R = 1|D)$. The words "optimal performance" refer both to informal user satisfaction (as long as redundancy is ignored) and to formal measures of evaluating IR performance. For example, it is not difficult to prove that using PRP to rank all documents in a given collection will lead to the greatest expected number of relevant hits in the top n ranks for every value of n (see [114]). Once that is established, it becomes obvious that the PRP maximizes *recall* and *precision* at rank n[1], as well as any measures that are derived from recall and precision, such as the F-measure [137], R-precision and mean average precision (MAP). Robertson [114] and Van Rijsbergen [137] also demonstrate that the Probability Ranking Principle leads to minimal decision-theoretic loss associated with retrieving a set of n top-ranked documents.

The probability ranking principle, as stated by Robertson is quite broad – it does not restrict us to any particular type of relevance, and the document representation D can be potentially very rich, covering not just the topical content of the document. In fact D could include features determining readability of the document, its relation to the user's preferences or suitability for a particular task. Similarly, R may well refer to "pertinence" or any other complex notion of relevance. The only restriction imposed by the PRP is that relevance of a particular document be scalar and independent of any other document in the collection. Consequently, PRP cannot handle issues of novelty and redundancy, or cases where two documents are relevant when put together, but irrelevant when viewed individually. Robertson [114] also cites a curious counter-example (due to Cooper) regarding the optimality of the principle. The counter-example considers the case when we are dealing with two classes of users who happen to issue the same request but consider different documents to be relevant to it. In that case PRP will only be optimal for the larger of the two user classes.

While PRP itself is quite general, in practice most probabilistic models take a somewhat more narrow view of it. In most cases the relevance R is

[1] *Recall* is defined as the number of relevant documents retrieved in the first n ranks, divided by the total number of relevant documents. *Precision* is the number of relevant documents in the first n ranks, divided by n.

restricted to mean *topical* relevance of a full-text document to a natural-language request. Relevance is also assumed to be fixed *a priori* in a form of relevance judgments for every document in the collection. These judgments are not directly observable by the system, but it is assumed that they exist and that they do not change in the course of a search session.

In the remainder of this section we will take a detailed look at several prominent probabilistic models of information retrieval. All of these models are either directly based on the PRP, or can be closely related to the principle. The major distinction between the models lies in how the authors choose to estimate the probability of relevance $P(R = 1|D)$.

2.3.2 The Classical Probabilistic Model

We will first consider a probabilistic model of retrieval proposed by Robertson and Sparck Jones in [117]. The model was initially named the *Binary Independence Model*, reflecting the basic assumptions it made about occurrences of words in documents. However, since 1976 the model has been re-formulated a number of times to a degree where it can hardly be called "binary" and, as we shall argue later on, the term "independence" is also questionable. The model is also known as the *Okapi model*, the *City model*, or simply as *the probabilistic model*. A very detailed account of the recent state of the model is provided by the original authors in [131, 132]. What follows is our interpretation of the model. An attentive reader may find that our description is different in two ways from the way the model is presented by the authors [117, 131, 132]. First, we choose to describe the model in terms of probabilities, as opposed to log-likelihood weights. Second, we make explicit some of the assumptions that are never stated by the authors, particularly in section 2.3.2. Both steps are taken to make the description more compatible with subsequent material.

Robertson and Sparck Jones start the development of their model by transforming the probability ranking principle into the rank-equivalent likelihood ratio:

$$\begin{aligned} P(R=1|D) &\overset{\text{rank}}{=} \frac{P(R=1|D)}{P(R=0|D)} \\ &= \frac{P(D|R=1)P(R=1)}{P(D|R=0)P(R=0)} \\ &\overset{\text{rank}}{=} \frac{P(D|R=1)}{P(D|R=0)} \end{aligned} \tag{2.1}$$

Here R is a Bernoulli random variable indicating whether or not a given document is relevant and D is a random variable representing the contents of that document. We assume that D takes values in some finite space $\mathcal{D}$, and that P represents a joint probability measure over $\{0, 1\} \times \mathcal{D}$. The first step of equation (2.1) comes from the fact that R is a binary variable and $\frac{P}{1-P}$ is a monotone (rank-preserving) transformation of P. The second step is a

straightforward application of Bayes' rule. In the third step we observe that the factor $\frac{P(R=1)}{P(R=0)}$ is a constant independent of D, and thus does not affect the ranking of documents. In order to proceed further we need to specify the nature of the document space $\mathcal{D}$. The space has to be flexible enough to capture the semantic content of the documents, and yet simple enough to allow estimation from limited data. Robertson and Sparck Jones take $\mathcal{D}$ to be the space of all subsets of the vocabulary $\mathcal{V}$ of our collection. Equivalently, a document D is a binary occurrence vector, such that $D_v = 1$ if word number v is present in the document, otherwise $D_v = 0$. The document space $\mathcal{D}$ is then the space of all possible binary vectors over the vocabulary: $\mathcal{D} = \{0,1\}^{N_\mathcal{V}}$, and the entire probability space of the model in question is $\{0,1\} \times \{0,1\}^{N_\mathcal{V}}$ – the same finite space as the space of $(N_\mathcal{V} + 1)$ coin tosses. As the next step in developing their model, Robertson and Sparck Jones assume that binary random variables D_i are mutually independent given the value of R, allowing them to factor the probabilities $P(D|R = 1)$ and $P(D|R = 0)$ as follows:

$$\begin{aligned}\frac{P(D{=}\mathbf{d}|R{=}1)}{P(D{=}\mathbf{d}|R{=}0)} &= \prod_{v\in\mathcal{V}} \frac{P(D_v{=}d_v|R{=}1)}{P(D_v{=}d_v|R{=}0)} \\ &= \left(\prod_{v\in D} \frac{p_v}{q_v}\right)\left(\prod_{v\notin D} \frac{1-p_v}{1-q_v}\right) \end{aligned} \tag{2.2}$$

The first step in equation (2.2) comes from assuming independence between random variables D_v. We honor the common practice of using capital letters (e.g. D_v) to denote random variables and lowercase letters (e.g. d_v) to refer to their observed values. The second step comes from the fact that D_v can only take values of 0 and 1, and using a shorthand $p_v = P(D_v{=}1|R{=}1)$ and $q_v = P(D_v{=}1|R{=}0)$. Also note the slight abuse of notation in the product subscripts: expression $v{\in}D$ really means $\{v{\in}\mathcal{V} : d_v{=}1\}$. As the final step Robertson and Sparck Jones desire that an empty document ($\mathbf{0}$) correspond to a natural zero in the log-space of their ranking formula. They achieve this by dividing equation (2.2) by $\frac{P(\mathbf{0}|R=1)}{P(\mathbf{0}|R=0)}$. Since that quantity is independent of any document, dividing by it will not affect document ranking and will yield the following final ranking criterion:

$$\begin{aligned} P(R{=}1|D{=}\mathbf{d}) &\propto \frac{P(D{=}\mathbf{d}|R{=}1)}{P(D{=}\mathbf{d}|R{=}0)} \Big/ \frac{P(D{=}\mathbf{0}|R{=}1)}{P(D{=}\mathbf{0}|R{=}0)} \\ &= \prod_{v\in D} \frac{p_v(1-q_v)}{q_v(1-p_v)} \end{aligned} \tag{2.3}$$

Parameter estimation with relevance information

Next comes the problem of estimation: in order to apply equation (2.3) to document retrieval we need to specify the quantities p_v and q_v, which reflect

the probabilities of the word v occurring in a relevant and a non-relevant document respectively. Robertson and Sparck Jones start with the case where the relevance variable R is observable, that is for every document D=**d** in the collection they know whether it is relevant or not. If that is the case, a natural estimate of p_v is the proportion of relevant documents that contain word v, and similarly q_v is the proportion of non-relevant documents containing v. However, when R is fully observable, there is really no point in ranking: we could simply return the documents for which $R = 1$. A more realistic case is when R is partially observable, i.e. for some documents we know whether they are relevant or not, for others R is unknown. This is precisely the environment in Information Filtering, Topic Tracking or Relevance Feedback tasks. For that case Robertson and Sparck Jones adjust the relative frequencies of v in the relevant and non-relevant documents by adding a constant 0.5 to all counts:

$$\begin{aligned} p_v &= \frac{N_{1,v} + 0.5}{N_1 + 0.5} \\ q_v &= \frac{N_{0,v} + 0.5}{N_0 + 0.5} \end{aligned} \tag{2.4}$$

Here N_1 is the number of known relevant documents, $N_{1,v}$ of them contain the word v. Similarly N_0 and $N_{0,v}$ reflect the total number of known non-relevant documents and how many of them contain v. The constant 0.5 in equation (2.4) serves two purposes: first, it ensures that we never get zero probabilities for any word v, and second, it serves as a crude form of smoothing (shrinkage), reducing the variance of estimates over possible sets of feedback documents.

Parameter estimation without relevance information

Until this point development of the Okapi model was quite straightforward. Unfortunately, it was based on the relevance variable R being at least partially observable, and that is simply not the case in a typical retrieval environment. In a typical scenario the only thing we have is the user's request Q, usually expressed as a short sentence or a small set of keywords. All our probability estimates have to be based on Q and on the collection as a whole, without knowing relevance and non-relevance of individual documents. To complicate matters further, Q is not even present in the original definition of the model (eqs. 2.1-2.3), it becomes necessary only when we have no way of observing the relevance variable R. Faced with these difficulties, Robertson and Sparck Jones make the following assumptions:

1. p_v=q_v if $v \notin Q$. When a word is not present in the query, it has an equal probability of occurring in the relevant and non-relevant documents. The effect of this assumption is that the product in equation (2.3) will only include words that occur both in the document and in the query, all other terms cancel out.

2. p_v=0.5 if $v \in Q$. If a word does occur in the query, it is equally likely to be present or absent in a relevant document. The assumption was originally proposed by Croft and Harper [34] and later re-formulated by Robertson and Walker [112]. The effect is that p_v and $(1 - p_v)$ cancel out in equation (2.3), leaving only $\frac{1-q_v}{q_v}$ under the product.
3. $q_v \propto N_v/N$. Probability of a word occurring in a non-relevant document can be approximated by its relative frequency in the entire collection. Here N is the total number of documents in the collection, N_v of them contain v. This approximation makes $\frac{1-q_v}{q_v}$ be proportional to the *inverse document frequency* (IDF) weight – a simple but devilishly effective heuristic introduced by Sparck Jones in [128].

Note that assumption (3) is quite reasonable: for a typical request only a small proportion of documents will be relevant, so collection-wide statistics are a good approximation to the non-relevant distribution. The same cannot be said for assumptions (1) and (2).

2-Poisson extension of the classical model

The original definition of the classical model deals exclusively with the binary representation of documents and queries, where a word is either present or not present in the document. However, empirical evidence suggests that the number of times a word is repeated within a document may be a strong indicator of relevance, and consequently the Okapi model was extended to include term frequency information. The first step in such extension is to expand the space $\mathcal{D}$ that is used to represent the documents. Previously, $\mathcal{D}$ was the set of all subsets of vocabulary $\{0,1\}^{N_\mathcal{V}}$. In order to handle frequencies, one can expand $\mathcal{D}$ to be $\{0,1,2\ldots\}^{N_\mathcal{V}}$. Now the ranking formula from equation (2.3) becomes:

$$\begin{aligned} P(R{=}1|D{=}\mathbf{d}) &\propto \frac{P(D{=}\mathbf{d}|R{=}1)}{P(D{=}\mathbf{d}|R{=}0)} \Big/ \frac{P(D{=}\mathbf{0}|R{=}1)}{P(D{=}\mathbf{0}|R{=}0)} \\ &= \prod_{v \in D} \frac{p_v(d_v) q_v(0)}{q_v(d_v) p_v(0)} \end{aligned} \tag{2.5}$$

Here d_v is the number of times word v was observed in the document. $p_v(d_v)$ is a shorthand for $P(D_v{=}d_v|R{=}1)$, the probability of seeing d_v occurrences of v in a relevant document, and $q_v(d_v)$ is the corresponding probability for the non-relevant document. Robertson and Sparck Jones [132] base their estimates of p_v and q_v on the 2-Poisson indexing model developed by Harter [52]. Harter's formalism revolves around a notion of *eliteness*, which was developed to model the behavior of a human indexer. Imagine a librarian who decides which keywords should be assigned to a given document. If he picks word v as a keyword for document d, then we say that d belongs to the *elite* class of v. Otherwise d belongs to the *non-elite* class. We would expect that documents

in the elite class of v are likely to contain many repetitions of v, while in the non-elite class v would primarily occur by chance. Harter assumed that frequency of v in both classes follows a Poisson distribution, but that the mean is higher in the elite class. Under this assumption, the frequency of v in the collection as a whole would follow a mixture of two Poissons:

$$P(D_v{=}d_v) = P(E{=}1)\frac{e^{-\mu_{1,v}}\mu_{1,v}^{d_v}}{d_v!} + P(E{=}0)\frac{e^{-\mu_{0,v}}\mu_{0,v}^{d_v}}{d_v!} \tag{2.6}$$

Here E is a binary variable specifying whether D is in the elite set of v, $\mu_{1,v}$ is the mean frequency of v in the elite documents, and $\mu_{0,v}$ is the same for the non-elite set. Since we don't know which documents are elite for a given word, we need some way to estimate three parameters: $\mu_{1,v}$, $\mu_{0,v}$ and $P(E{=}1)$. Harter's solution was to fit equation (2.6) to the empirical distribution of v in the collection using the method of moments. But eliteness is not quite the same as thing as relevance, since eliteness is defined for single words and cannot be trivially generalized to multi-word requests. In order to fit Harter's model into the Okapi model the authors had to make some adjustments. Robertson and Walker [118] proposed to condition eliteness on R, and assumed that once we know eliteness, the frequency of v in a document is independent of relevance:

$$p_v(d_v) = P(E{=}1|R{=}1)\frac{e^{-\mu_{1,v}}\mu_{1,v}^{d_v}}{d_v!} + P(E{=}0|R{=}1)\frac{e^{-\mu_{0,v}}\mu_{0,v}^{d_v}}{d_v!} \tag{2.7}$$

$$q_v(d_v) = P(E{=}1|R{=}0)\frac{e^{-\mu_{1,v}}\mu_{1,v}^{d_v}}{d_v!} + P(E{=}0|R{=}0)\frac{e^{-\mu_{0,v}}\mu_{0,v}^{d_v}}{d_v!} \tag{2.8}$$

Substituting equations (2.7,2.8) back into the ranking formula (2.5), leads to a rather messy expression with a total of 4 parameters that need to be estimated for every word v: the mean frequencies in the elite and non-elite sets ($\mu_{1,v}$ and $\mu_{0,v}$), and the probability of eliteness given relevance or non-relevance ($P(E{=}1|R{=}1)$ and $P(E{=}1|R{=}0)$). This presents a rather daunting task in the absence of any relevance observations, leading the authors to abandon formal derivation and resort to a heuristic. They hypothesize that equations (2.7,2.8) might lead to the following term under the product in equation (2.5):

$$\frac{p_v(d_v)q_v(0)}{q_v(d_v)p_v(0)} \approx \exp\left\{\frac{d_v \cdot (1+k)}{d_v + k \cdot \left((1-b) + b\frac{n_d}{n_{avg}}\right)} \times \log\frac{N}{N_v}\right\} \tag{2.9}$$

The quantity under the exponent in equation 2.9 represents the well-known and highly successful *BM25* weighting formula. As before, d_v is the number of times v occurred in the document, n_d is the length of the document, n_{avg} is the average document length in the collection, N is the number of documents in the collection and N_v is the number of documents containing v. k and b represent constants that can be tuned to optimize performance of the model on the task at hand. We stress the fact that equation (2.9) is not a

derived result and does not follow from any set of meaningful assumptions about the constituents of equations (2.7,2.8). *BM25* is a work of art, carefully engineered to combine the variables that were empirically found to influence retrieval performance: term frequency d_v, document length n_d and the inverse document frequency $\log\frac{N}{N_v}$. It is simple, flexible and very effective on a number of tasks (see [131, 132]). Unfortunately, it has no interpretation within the probabilistic model.

Modeling dependence in the classical model

The assumption of word independence in the classical model is a favorite target of linguistically sophisticated critics and aspiring graduate students. No other aspect of the formalism has drawn so much criticism and so many failed endeavors to improve the model[2]. Recall that the assumption states that individual words D_i in the document are mutually independent given the relevance variable R. The assumption is formalized in equation (2.2) for binary document representation and in equation (2.5) for the non-binary case. The assumption is intuitively wrong – we know that words in a language are not independent of each other: supposing that presence of the word "politics" tells us nothing about occurrence of "Washington" is clearly absurd. The popular perception is that the assumption of independence is a necessary evil, it is tolerated simply because without it we would have to estimate joint probabilities for vectors involving half a million random variables each (typical vocabulary size), and that is clearly intractable. Another popular perception is that there must be a way to partially model these dependencies, bringing the model closer to reality, and surely improving the retrieval performance.

One of the first attempts to relax the independence assumption is due to Van Rijsbergen [136]. The idea is to allow pairwise dependencies between words, such that for every word v there exists a *parent* word $\pi(v)$ which influences presence or absence of v. There is also a *root* word v_0 which has no parent. Dependencies form a spanning tree over the entire vocabulary, the structure of that tree can be induced automatically from a corpus by maximizing some objective function. Van Rijsbergen suggested using the aggregate mutual information over the branches ($\sum_v I(v, \pi(v))$) as the objective to maximize, other measures may work equally well. Once the structure $\pi(\cdot)$ is determined, we can replace the probabilities $P(D_v{=}d_v|R)$ in equations (2.2) and (2.5) with their conditional counterparts $P(D_v{=}d_v|D_{\pi(v)}{=}d_{\pi(v)}, R)$. After re-arranging the indices v in the products to descend down the tree, we have a way to model relevance without assuming mutual independence.

[2] It is a personal observation that almost every mathematically inclined graduate student in Information Retrieval attempts to formulate some sort of a non-independent model of IR within the first two to three years of his or her studies. The vast majority of these attempts yield no improvements and remain unpublished.

Unfortunately, empirical evaluations [51, 50] of the new model suggest that by and large it performs no better than the original. When improvements were observed they were mostly attributed to *expanding* the query with additional words, rather than to a more accurate modeling of probabilities. Disappointing performance of complex models is often blamed on combinatorial explosion of the number of parameters. However, in Van Rijsergen's model the total number of parameters is only twice that of the original formulation: we replace p_v in equation (2.2) with $p_{v,0}$ and $p_{v,1}$, reflecting absence and presence of $\pi(v)$; the same is done for q_v. This suggests that number of parameters may not be the culprit behind the lack of improvement in retrieval accuracy. Neither can we blame performance on the particular choices made in [136] – during the two decades that passed, Van Rijsbergen's idea has been re-discovered and re-formulated by various researchers in wildly different ways [26, 27, 64, 80, 110, 137, 156]. In most cases the results are disappointing: consistent improvement is only reported for very selective heuristics (phrases, query expansion), which cannot be treated as formal models of word dependence. The pattern holds both when relevance is not observable (ad-hoc retrieval) and when there are a lot of relevant examples (text classification). In the latter case even phrases are of minimal value.

Why dependency models fail

It is natural to wonder why this is the case – the classical model contains an obviously incorrect assumption about the language, and yet most attempts to relax that assumption produce no consistent improvements whatsoever. In this section we will present a possible explanation. We are going to argue that the classical Binary Independence Model really *does not* assume word independence, and consequently that there is no benefit in trying to relax the non-existent assumption. Our explanation is an extension of a very important but almost universally ignored argument made by Cooper in [29]. Cooper argues that in order to arrive at equation (2.2), we only need to assume *linked dependence* between words D_i, and that assumption is substantially weaker than independence. Cooper's argument is as follows. Consider the case of a two-word vocabulary $\mathcal{V}=\{a,b\}$, and suppose we do not assume independence, so $P_1(D_a, D_b)$ is their joint distribution in the relevant class, $P_0(D_a, D_b)$ is the same for the non-relevant class. Now for a given document $D=\mathbf{d}$, consider the following quantities:

$$k_1 = \frac{P_1(d_a, d_b)}{P_1(d_a)P_1(d_b)}$$
$$k_0 = \frac{P_0(d_a, d_b)}{P_0(d_a)P_0(d_b)} \tag{2.10}$$

By definition, k_1 is a measure of dependence between events $D_a{=}d_a$ and $D_b{=}d_b$ in the relevant class; it tells as how wrong we would be if we assumed

D_a to be independent of D_b. If $k_1>1$, the events d_a and d_b are positively correlated in the relevant class, $k_1<1$ means they are negatively correlated. k_0 plays the same role for the non-relevant class. Without assuming independence, the posterior odds of relevance (equation 2.1) takes the form:

$$\begin{aligned} P(R{=}1|D{=}\mathbf{d}) &\propto \frac{P(D{=}\mathbf{d}|R{=}1)}{P(D{=}\mathbf{d}|R{=}0)} \\ &= \frac{P_1(d_a,d_b)}{P_0(d_a,d_b)} \\ &= \frac{k_1 P_1(d_a)P_1(d_b)}{k_0 P_0(d_a)P_0(d_b)} \end{aligned} \tag{2.11}$$

When Robertson and Sparck Jones [117] assume that D_a and D_b are independent, they in effect assume that $k_1{=}1$ and $k_0{=}1$. But Cooper [29] correctly points out that to justify equation (2.2) we only need to assume $k_1{=}k_0$, which is much less restrictive: k_1 and k_2 can equal any number, not just 1. This is Cooper's *linked dependence* assumption, it demonstrates that the classical model actually allows for any degree of dependence between words a an b, as long as that dependence is exactly the same in the relevant and non-relevant classes.

Cooper's assumption is certainly more reasonable than mutual independence, but it has its limitations. For example, if the user's request happens to deal with compound concepts, such as "machine translation", it would be disastrous to assume the same degree of dependence for these two words in the relevant and non-relevant documents. Additionally, linked dependence presented by Cooper, becomes more and more restrictive as we consider larger vocabularies and deal with factors k_1 and k_0 of the form $\frac{P(d_1 \ldots d_n)}{P(d_1)\ldots P(d_n)}$. However, we would like to argue that Cooper's argument can be taken one step further, yielding an even weaker assumption that can withstand the counterexample given above. We will refer to this as the assumption of **proportional interdependence**.Let $\mathcal{V}$ be a general vocabulary. As a first step, we will restrict our discussion to the simple case where only first-order dependencies exist between the words: a word v may only depend on one other word, as in Van Rijsbergen's model [136]. We will go a step further and allow each word v to have potentially different parents $\pi_1(v)$ and $\pi_0(v)$ in the relevant and non-relevant dependence trees. We know that under a first-order model, the joint distribution $P(D{=}\mathbf{d}|R{=}r)$ decomposes into a product of conditional probabilities $P(D_v{=}d_v|D_{\pi_r(v)}{=}d_{\pi_r(v)}, R{=}r)$, one for each word v in the vocabulary. Inspired by Cooper, we define the factor $k_{v,r}$ to be the ratio of the conditional probability to the unconditional one:

$$k_{v,r} = \frac{P(D_v{=}d_v|D_{\pi_r(v)}{=}d_{\pi_r(v)}, R{=}r)}{P(D_v{=}d_v|R{=}r)} \tag{2.12}$$

Now the version of equation (2.1) appropriate for a first-order dependence model will take the following form:

$$P(R{=}1|D{=}\mathbf{d}) \propto \prod_{v\in\mathcal{V}} \frac{P(D_v{=}d_v|D_{\pi_1(v)}{=}d_{\pi_1(v)}, R{=}1)}{P(D_v{=}d_v|D_{\pi_0(v)}{=}d_{\pi_0(v)}, R{=}0)}$$
$$= \prod_{v\in\mathcal{V}} \frac{P(D_v{=}d_v|R{=}1)}{P(D_v{=}d_v|R{=}0)} \cdot \frac{k_{v,1}}{k_{v,0}} \tag{2.13}$$

Equation (2.13) makes it very clear that the first-order model is rank-equivalent to the independence model if and only if $\prod_v \frac{k_{v,1}}{k_{v,0}}$ is a constant independent of $\mathbf{d}$. An equivalent statement is that cumulative pairwise mutual information between the presence of a randomly picked word and the presence of its parent differs by a constant k (independent of $\mathbf{d}$) in the relevant and non-relevant classes:

$$\sum_{v\in\mathcal{V}} \log \frac{P_1(d_v, d_{\pi_1(v)})}{P_1(d_v)P_1(d_{\pi_1(v)})} = k \sum_{v\in\mathcal{V}} \log \frac{P_0(d_v, d_{\pi_0(v)})}{P_0(d_v)P_0(d_{\pi_0(v)})} \tag{2.14}$$

Informally, equation (2.14) means that *on average*, all the words in a given document have about as much interdependence under the relevant class (P_1) as they do under the non-relevant class (P_0). The key phrase here is "on average": equation (2.14) does not require that any two words be equally dependent under P_1 and P_1 – that is precisely Cooper's linked dependence. Instead, equation (2.14) allows some words to be strongly dependent only in the relevant class (e.g., "machine" and "translation"), as long as on average they are balanced out by some dependencies that are stronger in the non-relevant class. They don't even have to balance out exactly ($k{=}0$), the only requirement is that whatever disbalance exists be constant across all documents.

We view the above as a strong result: the independence model is equivalent to any first-order dependence model under a very weak **proportional interdependence** assumption, that we personally believe holds in most situations.[3] Indeed, we see no reason to believe that an arbitrary set of documents in the collection (the relevant set for some request) will exhibit a stronger cumulative dependence over all words than will the complement of that set. The meaning of this result is that any attempt to replace independence with first-order dependence is very likely to produce no improvements, other than by

[3] If desired, one could certainly test the empirical validity of the proportional interedependence assumption for a given collection. The test would proceed as follows. **(1)** partition the collection into relevant and non-relevant sets using complete relevance judgments. **(2)** estimate the dependency structures $\pi_1()$ and $\pi_0()$ for the relevant and non-relevant classes (e.g., using Van Rijsbergen's method). **(3)** construct maximum-likelihood estimates for conditional distributions $P_r(v|\pi_r(v)) : r{\in}\{0,1\}$. **(4)** compute the value k in equation (2.14) for each document d in the collection. **(5)** perform a statistical goodness-of-fit test, comparing the set of values k observed for the relevant documents against the values observed for non-relevant documents. If the null hypothesis (that the populations are identical) cannot be rejected, then the proportional interdependence assumption holds for this collection.

accident. We also point out that this result may not be limited to first-order dependencies. One could define factors $k_{v,0}$ and $k_{v,1}$ for higher-order models where word probabilities are conditioned on *neighborhoods* $\eta(v)$ instead of of *parents* $\pi(v)$. Admittedly, the argument becomes somewhat more elaborate; we have not worked out the details. As a conclusion to this section, we would like to stress the following:

> Contrary to popular belief, word independence is not a necessary assumption in the classical probabilistic model of IR. A necessary and sufficient condition is proportional interdependence, which we believe holds in most retrieval settings. If there is anything wrong with the classical model, it is not independence but the assumptions made in the estimation process (see sections 2.3.2 and 2.3.2).

2.3.3 The Language Modeling Framework

We will now turn our attention to a very different approach to relevance – one based on statistical models of natural language. Statistical language modeling is a mature field with a wide range of successful applications, such as discourse generation, automatic speech recognition and statistical machine translation. However, using language modeling in the field of Information Retrieval is a relatively novel development. The approach was proposed by Ponte and Croft [106, 105] in 1998, and in the short time since then it has attracted a tremendous level of interest and a growing number of publications each year. In this section we will outline the original language modeling approach to IR [106] and briefly mention some of the more prominent extensions.

One of the main motivations Ponte and Croft had for developing the language modeling approach was to get away from the heuristics that came to dominate the probabilistic model of Robertson and Sparck Jones [117]. Recall that heuristics in the classical model arise when we are given no examples to estimate the probabilities p_v associated with relevant documents. Ponte and Croft's solution to this problem was quite radical – it was to remove the explicit relevance variable R, and construct a probabilistic model around the document and the user's query. The authors hypothesized that for every document D=$\mathbf{d}$, there exists an underlying language model M_d. Now, if the query Q=$\mathbf{q}$ looks like it might be a random sample from M_d, we have a reason to believe that $\mathbf{d}$ is relevant to $\mathbf{q}$. Informally, we can think of M_d as a crude model reflecting the state of mind of the author who created document $\mathbf{d}$. If the same state of mind is likely to produce the query $\mathbf{q}$, then it is likely that $\mathbf{q}$ is topically related to $\mathbf{d}$, hence $\mathbf{d}$ would be topically relevant.

The effect of Ponte and Croft's argument is that they could replace the probability of relevance $P(R{=}1|D{=}\mathbf{d})$ with the probability of observing the query from the language model of the document $P(Q{=}\mathbf{q}|M_d)$. This is a crucial step: it allows the authors to avoid the uncertainty associated with the unobserved relevance variable R. Indeed, Q is observable, and there exists a

substantial body of statistical literature to help us in estimating M_d from the observed document $\mathbf{d}$. Retrieval in Ponte and Croft's model can be decomposed into two steps. First, we have to use the observation $\mathbf{d}$ to construct our estimate of the underlying document language model M_d. Second, we can compute the probability $P(Q{=}\mathbf{q}|M_d)$ of observing $\mathbf{q}$ as a random sample from M_d, and rank all documents in the decreasing order of that probability.

Multiple-Bernoulli language models

Ponte and Croft represent queries in the same space that was used by Robertson and Sparck Jones in the Binary Independence Model. If $\mathcal{V}$ is a vocabulary of $N_\mathcal{V}$ words, the query space $\mathcal{Q}$ is the set of all subsets of vocabulary ($\{0,1\}^{N_\mathcal{V}}$). The query Q is a vector of $N_\mathcal{V}$ binary variables Q_v, one for each word v in the vocabulary. The components Q_i are assumed to be mutually independent conditioned on the language model M_d. The language model itself is a vector of $N_\mathcal{V}$ probabilities $p_{d,v}$, one for each word v. The probability of observing a query $Q{=}\mathbf{q}$ from a given model $M_d{=}\mathbf{p}_d$ is:

$$\begin{aligned} P(Q{=}\mathbf{q}|M_d{=}\mathbf{p}_d) &= \prod_{v\in\mathcal{V}} P(Q_v{=}q_v|M_d{=}\mathbf{p}_d) \\ &= \prod_{v\in Q} p_{d,v} \times \prod_{v\notin Q}(1-p_{d,v}) \end{aligned} \tag{2.15}$$

Here again, $v{\in}Q$ is a shorthand for $\{v{\in}\mathcal{V} : q_v{=}1\}$, and likewise for the complement set. Ponte and Croft propose the following way to compute $\mathbf{p}_d$ from the document $\mathbf{d}$:

$$p_{v,d} = \begin{cases} \left(\frac{d_v}{|\mathbf{d}|}\right)^{(1-r)} \left(\frac{1}{N_v}\sum_{\mathbf{d}'}\frac{d'_v}{|\mathbf{d}'|}\right)^{r} & \text{if } d_v > 0 \\ \left(\sum_{\mathbf{d}'} d'_v\right) / \left(\sum_{\mathbf{d}'} |\mathbf{d}'|\right) & \text{otherwise} \end{cases} \tag{2.16}$$

Here d_v is the number of times word v occurs in document $\mathbf{d}$, $|\mathbf{d}|$ denotes the length of document $\mathbf{d}$, N_v is the number of documents containing v and the summations go over every document $\mathbf{d}'$ in the collection. If a word v does not occur in the document, Ponte and Croft use its relative frequency in the entire collection. If a word does occur, the estimate is a weighted geometric average between the relative frequency in the document and the average relative frequency over all documents containing v. The weight is given by the parameter r, which according to the authors plays the role of *Bayesian risk*.

A probabilist will undoubtedly notice an inconsistency between equations (2.15) and (2.16). The former represents a *multiple-Bernoulli* distribution over the binary event space, but the probability estimates in equation (2.16) are based on non-binary frequencies d_v and would naturally arise if we assumed that $\mathbf{d}$ was a random sample from a *multinomial* distribution. Ponte and Croft never address the issue, but elsewhere [88] we show that the model can be made consistent by assuming that each document $\mathbf{d}$ is represented not by a single set of words, but by a set of $|\mathbf{d}|$ singleton sets, each assumed to be independently drawn from M_d.

Multinomial language models

As we mentioned above, term frequencies are somewhat unnatural in Ponte and Croft's model. The model is explicitly geared to capture the presence or absence of words, and does not recognize the fact that words can be repeated in the query. This is perfectly reasonable for short 2-3 word queries that are typical of web searches, but it is not a good assumption for the general retrieval setting. In order to take account of frequencies researchers have had to assume a different event space. Virtually every publication concerning language modeling in IR [126, 127, 90, 11, 152, 58, 56, 158, 159] presumes the following representation, though it is rarely stated in formal terms. Assume $\mathcal{V}$ is a vocabulary of $N_\mathcal{V}$ distinct words. Both documents and queries are viewed as strings (sequences) over $\mathcal{V}$. A document D of length m is a sequence of m random variables D_i, each taking values in the vocabulary $\mathcal{V}$. The query Q has the same representation: $Q{=}n, Q_1 \ldots Q_n$, such that $Q_i{\in}\mathcal{V}$ for each $i = 1 \ldots n$. The probability space for both documents and queries is the space of all possible sequences of words: $\mathcal{Q} = \mathcal{D} = I\!N \times \cup_{n=1}^{\infty} \mathcal{V}^n$. Note: most authors omit sequence length N from the representation. We make it explicit to define a single probability measure for strings of any length. Individual words Q_i in the sequence are assumed to be independent of N, independent of each other and identically distributed according to the language model M_d. M_d now plays the role of a discrete distribution over the vocabulary; its values are vectors of $N_\mathcal{V}$ probabilities, one for each word v: $\mathbf{p}_d{\in}[0,1]^{N_\mathcal{V}}$ such that $1{=}\sum_v \mathbf{p}_{d,v}$. The probability mass assigned by a language model M_d to some string Q is:

$$\begin{aligned} P(Q{=}\mathbf{q}|M_d{=}\mathbf{p}_d) &= P(N{=}n) \prod_{i=1}^{n} P(Q_i{=}q_i|M_d{=}\mathbf{p}_d) \\ &= \pi_n \prod_{i=1}^{n} p_{d,q_i} \end{aligned} \tag{2.17}$$

Here $P(N{=}n) = \pi_n$ is some discrete distribution over string lengths; it is independent of everything else and is usually assumed to be uniform until some maximum length M and zero beyond that. A common way to estimate the language model is to assume that the document $\mathbf{d}$ itself represents a random sample drawn from M_d, and use relative frequencies of words in $\mathbf{d}$ as a maximum likelihood estimate $\mathbf{p}_d$. However, maximum likelihood estimation will naturally lead to zeros in the estimate, so some form of smoothing is required. From the large pool of available smoothing techniques [23, 158], most authors pick some form of linear interpolation between the maximum likelihood estimate and the "background" frequency of a word computed over the whole collection:

$$p_{d,v} = \lambda_d \frac{n_{d,v}}{n_d} + (1 - \lambda_d) \frac{n_{c,v}}{n_c} \tag{2.18}$$

Here $n_{d,v}$ refers to the number of times the word v occurs in document $\mathbf{d}$, n_d is the length of $\mathbf{d}$, $n_{c,v}$ is the frequency of v in the entire collection and

n_c is the total number of words in the collection. Parameter λ_d is used to control the degree of variance in the estimator. Lafferty and Zhai [68] show that equation (2.18) is a natural Bayesian estimate that follows from assuming a Dirichlet prior with parameters proportional to $\frac{n_{c,v}}{n_c}$ over the simplex of all possible language models.

Multinomial and multiple-Bernoulli event spaces

As we already mentioned, the original language modeling framework proposed by Ponte and Croft [106] is defined over a binary event space, and does not allow for word repetitions. The multinomial approach described in section 2.3.3 does allow word frequencies, but that is not the only difference between the two frameworks. A much more important, and commonly overlooked difference is that the two approaches have very different and incompatible event spaces. The random variable Q means two completely different things in equations (2.15) and (2.17), and the meanings cannot be used interchangeably, or mixed together as was done in several publications. In Ponte and Croft's model, Q is a *vector* of binary variables Q_v. Each Q_v represents a *word* in the vocabulary, possible values of Q_v are 0 and 1 (absent and present). In the multinomial framework, Q is a sequence of variables Q_i, each Q_i represents a *position* in the query, and the values of Q_i are words. Note that the latter representation is absolutely not a requirement if we just want to model counts. We could have simply extended the range of Q_v in the Ponte and Croft model to include counts, as was done in the 2-Poisson extension of the Binary Independence Model. Doing this would give us a vector representation that is very different from the sequence representation described in section 2.3.3. To be specific, in the vector representation we have half a million random variables, each with two possible values: absent or present. In the sequence we effectively have only one variable (since Q_i are i.i.d.), but that variable can take half a million possible values.

Which of these representations is more suitable for Information Retrieval is a very interesting open question. Our feeling is that vector representation might be more natural. It allows us to estimate a separate distribution for every vocabulary word, makes no a-priori assumptions about word counts and allows us to explicitly model dependencies between different vocabulary words (though in light of section 2.3.2 we might wonder if dependencies are of any use at all). On the other hand, the sequence representation makes it more natural to model word proximity, phrases and other surface characteristics of text. In practice, the question of representation is all but settled – nearly every publication assumes sequences rather than vectors. This choice is largely a matter of consistency – in the fields adjacent to IR, such as Speech Recognition (ASR), Machine Translation (MT) and Natural Language Processing (NLP), language modeling is always concerned with sequences. In addition, a large body of language modeling publications in these fields serves as a gold-mine of estimation techniques that can be applied in IR – anything from n-gram

and cache models in ASR, to translation models in MT, to grammars in NLP. For the remainder of this book, when we speak of language models we will refer to sequence models, as defined in section 2.3.3.

Independence in the language modeling framework

Just like the classical probabilistic model, the language modeling approach relies on making a very strong assumption of independence between individual words. However, the meaning of this assumption is quite different in the two frameworks. In the classical model, independence means that presence of "politics" does not affect presence of "Washington". In the language modeling framework, the independence assumption means that the identity of the n'th word in the sequence does not depend on any preceding or following words. The second assumption implies the first: "Washington" in position n is still not affected by "politics" in position m. However the converse is not true: in addition to the above, the language modeling framework assumes that "politics" in position m does not affect "politics" in position n, so in a sense a word is independent of itself. This is certainly not the case in the classical model, although assuming a Poisson distribution in section 2.3.2 is essentially equivalent, since Poisson sampling is a memory-less process.

Given that the assumption of independence is even stronger in LM than in the classical model, it should come as no surprise that several researchers attempted to relax the assumption. One of the first attempts is due to Song and Croft [126, 127]. In that work each query word Q_i was conditioned on the immediately preceding Q_{i-1}, forming a first-order Markov Chain. The same assumption was made about the documents, and consequently the language model M_d takes the form of a bigram model. The parameters were estimated using bigram frequencies in the document with back-off to the unigram and to collection-wide counts. Note that in this case one does experience a combinatorial explosion: the number of parameters is squared, not just doubled as in Van Rijsbergen's dependence model [136]. As might be expected, the new model did not yield consistent improvements over the original formulation. A similar model was proposed by Miller and colleagues [89, 90], but bigram performance was never evaluated. A very different formalism was recently attempted by Nallapati [94, 95], who tried to combine Van Rijsbergen's dependence model [136] with the multinomial model in section (2.3.3). The results were inconsistent. We are also aware of several unpublished studies, where different attempts to introduce dependencies between random variables Q_i or D_i did not lead to any improvements over the unigram model.

Faced with the poor performance of dependency models, can we repeat the argument made in section 2.3.2 and show that document ranking is not affected by the assumption of independence? Our intuition is that we cannot: the classical probabilistic model involved only two probability distributions (relevant and non-relevant), while in the language-modeling framework we are faced with a distinct probability distribution M_d for each document $\mathbf{d}$

in the collection. Furthermore, interdependencies accumulate only over the query words, not over the entire vocabulary as was the case with the classical model. We see no reason to believe that aggregate dependence among the query words will not heavily depend on M_d. However, we will provide an informal argument for why modeling dependence does not seem to help in IR, whereas it is absolutely essential in other fields that use language models (ASR, MT, NLP). The primary use of language models in fields other than IR is to ensure surface consistency, or well-formedness of strings of words. As an example, consider a speech recognition system, which typically consists of an acoustic model and a language model. To an acoustic model the utterance "I see defeat" may appear no different from a nonsensical "icy the feet". But any decent bigram model would favor the first string as more consistent. Similarly, in NLP a discourse generation system may use a grammar of English to translate a template-based representation of some action into a well-formed sentence. In these fields it is absolutely necessary to worry about the surface form of strings because the goal is to generate *new* strings of text in response to some input. If the system generates gibberish, it is useless. Information retrieval is different in the sense that it deals with *existing* strings of text, which are already well-formed. When we directly adopt an n-gram model from speech recognition or a maximum-entropy model from MT, we are in effect adopting a proven solution for a problem that we do not face. At the same time, the added complexity of the new model will likely translate to less reliable parameter estimates than the ones we had with the simpler model.

The above argument should be taken with a grain of salt. We are not suggesting that it is impossible to improve the dependency structure of the language modeling approach. We only claim that no improvement should be expected from *direct* models of dependence – models where random variables Q_i are *directly* conditioned on some $Q_{j\neq i}$. That, however, does not mean that no improvement will result from models that capture dependencies *indirectly*, perhaps via a hidden variable. For example, Berger and Lafferty [11] proposed to model information retrieval as statistical translation of a document into the query. In the simplest instantiation of their approach (model 1), latent words T_i are randomly sampled from the document model M_d, and then probabilistically *translated* into query words Q_i according to some distribution $P(Q_i|T_i)$. Under the translation model, the document ranking criterion (equation 2.17) takes the following form:

$$\begin{aligned} P(Q{=}\mathbf{q}|M_d{=}\mathbf{p}_d) &\propto \prod_{i=1}^{n} \sum_{v\in\mathcal{V}} P(Q_i{=}q_i|T_i{=}v)P(T_i{=}v|M_d{=}\mathbf{p}_d) \\ &= \prod_{i=1}^{n} \sum_{v\in\mathcal{V}} t_{v,q_i} p_{d,v} \end{aligned} \tag{2.19}$$

In the translation model there is no direct dependency between the query words Q_i or the document words D_i. Instead, the translation matrix $t_{v,q}$ provides a useful way to handle dependencies between *identities* of the words in

the document and in the query. For instance, the translation model would correctly model dependency between the word "cat" in a document and the word "feline" in the query, if the translation matrix $t_{v,q}$ was estimated to reflect synonymy. The translation model was originally proposed as a general-purpose model of IR, but it found its greatest application in the field of cross-language retrieval, where documents in one language are queried using some other language. Two other prominent examples of the indirect approach to word dependencies are the latent aspect model of Hoffman [58], and the Markov-chain model proposed by Lafferty and Zhai [68].

2.3.4 Contrasting the Classical Model and Language Models

In the previous section we described two probabilistic frameworks for modeling relevance in information retrieval. One grew out of the Binary Independence Model, proposed by Robertson and Sparck Jones [117]; the other represents various developments of the language-modeling approach pioneered by Ponte and Croft [106]. This section will be devoted to looking at the benefits and drawbacks of the two frameworks and will serve as a bridge leading into the development of our own generative model of relevance.

Despite their fundamental differences, there are a number of similarities between the classical model and language models. Both approaches started with a focus on binary presence or absence of words, and, curiously, used exactly the same event space. Both rapidly evolved from binary occurrence to modeling word frequencies – one explicitly, via a Poisson distribution over the counts, the other, implicitly, by adopting a multinomial distribution over the vocabulary. Both frameworks appear to make a very strong assumption of independence, though we have argued that the meaning of *independence* is quite different in the two models. In the former, independence concerns word *identities* (presence of "politics" unaffected by "Washington"); in the latter word *positions* are assumed independent (first query/document word does not affect second word). Since both assumptions appear obviously incorrect, a lot of effort went into improving performance by modeling dependencies. Unfortunately, explicit models of dependence did not lead to consistent performance improvements in either framework. In an attempt to understand this curious effect, we provided two different arguments for why dependency models do not help in the classical framework and the language modeling framework. For the classical case we extended an argument initiated by Cooper [29] and showed that the model really does not assume independence, it is based on a much weaker assumption of proportional interdependence (see section 2.3.2 for details). For the language modeling framework we informally argued that explicit models of dependence will capture nothing but the surface form (well-formedness) of text, which has little to do with the topical content.

Relevance in the two frameworks

The point where language models become very different from the classical models is on the issue of relevance. The classical probabilistic model is centered around relevance: Robertson and Sparck Jones [117] start the development of their model directly from the probability ranking principle and proceed formally as far as they can (as long as relevance is observable). To contrast that, in the language modeling approach there is no explicit concept of relevance. Ponte and Croft [106] replace it with a simple generative formalism: probability of relevance $P(R|D)$ is assumed to be proportional to the probability of randomly drawing the query Q from the document model M_d. There is a clear benefit to this assumption: since both the query and the document are always fully observable, the model does not have to deal with the ambiguous concept of relevance. Ponte and Croft effectively turned a retrieval problem into an estimation problem. Instead of trying to model relevance we look for the best way to estimate the language model M_d for a given document d. This estimation step can be carried out in a systematic fashion, without resorting to heuristics that become necessary in the classical model.

However, absence of relevance raises the question of what to do in the rare cases when we do have relevant examples. To elaborate, suppose we have a small set of documents that are known to be relevant to a given query. How could we make use of this information to improve ranking of subsequent documents? The process is very straightforward in the classical probabilistic model – we simply use relative frequencies of words to get better probability estimates; the exact formulae are given in section 2.3.2. This can be done with both positive and negative (non-relevant) examples. With the language modeling framework incorporating relevance judgments is not nearly as clear. We cannot update probability estimates because there is no distribution associated with the relevant class. Updating the document model M_d makes no sense, since examples are relevant to the query, not to the document. Ponte [105] suggests that the only thing we can do is re-formulate the query, expanding it with words selected from relevant examples according to a heuristic weighting formula. Interestingly, language models allow for a very different kind of feedback that cannot be handled within the classical model. If for a given document **d** we have examples of relevant queries (queries for which **d** was judged relevant), we can certainly make use of those queries in adjusting the language model M_d. This form of feedback has recently been studied in [96].

Formal absence of relevance from the language modeling approach has also led to continued criticism of the framework [129, 111, 130]. Quoting from [130]: *"a retrieval model that does not mention relevance appears paradoxical"*. Responding to this criticism, Lafferty and Zhai [69] claim that the language modeling approach can be re-formulated to include a concept of relevance, albeit implicitly. To support their claim, Lafferty and Zhai argue that from a high-level viewpoint both frameworks operate with three random variables:

the query Q, the document D and the relevance variable R. Both frameworks attempt to approximate the posterior distribution over R, but the factorization of dependencies is done in different ways. In the classical probabilistic model, D is factored out first and conditioned on R and Q, leading to the familiar development:

$$\begin{aligned}\frac{P(R{=}1|D,Q)}{P(R{=}0|D,Q)} &= \frac{P(D|R{=}1,Q)}{P(D|R{=}0,Q)} \cdot \frac{P(R{=}1,Q)}{P(R{=}0,Q)} \\ &\propto \frac{P(D|R{=}1,Q)}{P(D|R{=}0,Q)}\end{aligned} \tag{2.20}$$

The leftmost portion of Figure 2.1 shows a graphical diagram of the dependencies implied by equation (2.20). Following convention, we use shaded circles to represent observed variables D and Q. A corresponding dependency diagram for the language modeling approach is shown in the middle of Figure 2.1. We use dashed lines to indicate that R is introduced somewhat artificially. The diagram results from factoring Q conditioned on R and D as follows:

$$\begin{aligned}\frac{P(R{=}1|D,Q)}{P(R{=}0|D,Q)} &= \frac{P(Q|R{=}1,D)}{P(Q|R{=}0,D)} \cdot \frac{P(R{=}1,D)}{P(R{=}0,D)} \\ &\propto P(Q|R{=}1,D) \cdot \frac{P(R{=}1,D)}{P(R{=}0,D)}\end{aligned} \tag{2.21}$$

In order to justify the second step above, Lafferty and Zhai have to assume that Q is independent of D if $R{=}0$, which means that the denominator of the first ratio does not affect the ranking and can be omitted. However, equation (2.21) still includes the ratio of joint probabilities over R and D, which is not present in the language modeling approach. To get rid of it, Lafferty and Zhai proposed to make R independent of D. This would leave $P(Q|R{=}1,D)$ as the only term that affects the ranking of documents, thus explaining the language modeling framework.

As a final note, we would like to suggest that there is a third way of factoring the joint distribution over R, D and Q. We could assume that the query Q and the document D are conditioned on the relevance variable R, and that Q and D are independent given R. This factorization is shown in the rightmost diagram in Figure 2.1, it forms the foundation for the model proposed in this book and will be discussed in great detail in the following chapter.

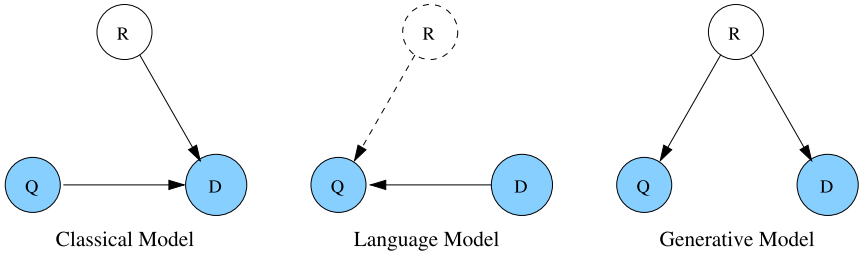

Fig. 2.1. Graphical diagrams showing dependencies between the query Q, the document D and relevance R variables in different probabilistic models of IR. Left: classical probabilistic model [117]. Middle: language modeling framework [106] according to [69]. Right: the generative model proposed in this book. Shaded circles represent observable variables.

http://www.springer.com/978-3-540-89363-9

A Generative Theory of Relevance

Lavrenko, V.

2009, XX, 197 p. 31 illus., Hardcover
ISBN: 978-3-540-89363-9