1a.

The Formula for Dirichlet Prior Smoothing is given by

$$P(w / d) = \frac{|d|}{|d| + \mu} P_{ML}(w / d) + \frac{\mu}{|d| + \mu} P(w / C)$$

Let us consider we have a very large document and |d| tends to infinite.

Before that let's refactor equation by dividing the numerator and denominator by |d|, we have

$$P(w / d) = \frac{1}{1 + \frac{\mu}{|d|}} P_{ML}(w / d) + \frac{\frac{\mu}{|d|}}{1 + \frac{\mu}{|d|}} P(w / C)$$

Since we have |d| tending to infinity and we know 1/x at x tends to infinity is zero substituting it we get

$$P(w / d) = \frac{1}{1 + 0} P_{ML}(w / d) + \frac{0}{1 + 0} P(w / C)$$

$$\mathbf{P(w / d) = P_{ML}(w / d)}$$

In a similar way when μ tends to infinity, we have the equation after dividing the numerator and denominator with μ we have

$$P(w / d) = \frac{\frac{|d|}{\mu}}{\frac{|d|}{\mu} + 1} P_{ML}(w / d) + \frac{1}{\frac{|d|}{\mu} + 1} P(w / C)$$

Since we have μ tending to infinity and we know 1/x at x tends to infinity is zero substituting it we get

$$P(w / d) = \frac{0}{0 + 1} P_{ML}(w / d) + \frac{1}{0 + 1} P(w / C)$$

P(w / d) = P(w / C)

1b.

Katz smoothing applies Good-Turing estimates to the problem of backoff language models.Katz smoothing uses a form of discounting in which the amount of discounting is proportional to that predicted by the Good-Turing estimate. Katz smoothing for higher-order n-grams is defined analogously. Katz smoothing doesn't consider any background model to estimate the terms rather it uses the previous n-1 gram to generate the n-gram. Since Katz is a generative model, the probability obtained may not be accurate, Jelinek-Mercer is better in this aspect since it uses background model which is better accurate. Ability to use Background Knowledge is an advantage.

2a.

Given document d is d: "the sun rises in the east and sets in the west"

The length of the document is |d| = 11

We know that from Unigram Language Model $P_{ML}(w / d) = \frac{c(w,d)}{|d|}$ using this we get

Word	$P_{ML}(w / d)$
a	0
the	3/11

from	0
retrieval	0
sun	1/11
rises	1/11
in	2/11
BM25	0
east	1/11
sets	1/11
west	1/11
and	1/11

The Formula for Dirichlet Prior Smoothing is given by

$$P(w / d) = \frac{|d|}{|d| + \mu} P_{ML}(w / d) + \frac{\mu}{|d| + \mu} P(w / C)$$

Using this formula we get

Word	P_{ML}(w / d)	*P(w / C)*	*P(w /d) for* $\mu = 4$
a	0	0.18	0.048
the	3/11	0.17	0.2453
from	0	0.13	0.0346
retrieval	0	0.02	0.0053
sun	1/11	0.05	0.08
rises	1/11	0.04	0.0773
in	2/11	0.16	0.176
BM25	0	0.01	0.00266
east	1/11	0.02	0.072
sets	1/11	0.04	0.0773
west	1/11	0.02	0.072
and	1/11	0.16	0.1093

2b.

Using the same approach for $\mu = 0.01$ and $\mu = 100$ we get

Word	P_{ML}(w / d)	*P(w / C)*	*P(w /d) for* $\mu = 0.01$	*P(w /d) for* $\mu = 100$
a	0	0.18	0.000163	0.1621
the	3/11	0.17	0.27263	0.1801
from	0	0.13	0.00011	0.1171
retrieval	0	0.02	0.0000181	0.0180
sun	1/11	0.05	0.09087	0.0540
rises	1/11	0.04	0.09086	0.0450
in	2/11	0.16	0.181	0.1621
BM25	0	0.01	0.0000090	0.00900
east	1/11	0.02	0.09084	0.02702
sets	1/11	0.04	0.09086	0.0450
west	1/11	0.02	0.09087	0.0270
and	1/11	0.16	0.09097	0.1531

In 2(a) μ is significant compared to d so the Background model has affect on the probabilities and the distribution is normalized, the value of P_{ML} is dominant since d>μ. In case of 2(b) when μ= 0.01

here μ is almost tending to zero and is a small value and hence the resulting probability will be almost equal to P_{ML} I.e P(w / d) $\cong P_{ML}(w / d)$. In the second case where μ =100, here μ is very large compared to d and it has complete dominance, hence in this case P(w / d) $\cong P(w / C)$.

2c.

We know that Jelinek-Mercer smoothing is given by the formula

P(w / d) = (1 - λ) $P_{ML}(w / d)$ + λ P(w / C)

Word	P_{ML}(w / d)	*P(w / C)*	*P(w /d) for* $\lambda = 0.01$	*P(w /d) for* $\lambda = 0.5$	*P(w /d) for* λ *=0.9*
a	0	0.18	0.0018	0.09	0.162
the	3/11	0.17	0.2717	0.221	0.1802
from	0	0.13	0.0013	0.0013	0.117
retrieval	0	0.02	0.0002	0.01	0.018
sun	1/11	0.05	0.0905	0.07045	0.054
rises	1/11	0.04	0.0904	0.0654	0.045
in	2/11	0.16	0.1816	0.17090	0.1621
BM25	0	0.01	0.0001	0.005	0.009
east	1/11	0.02	0.0902	0.0554	0.02709
sets	1/11	0.04	0.0904	0.0654	0.0450
west	1/11	0.02	0.0902	0.0554	0.0270
and	1/11	0.16	0.0916	0.1254	0.1530

When λ is low as 0.01 the resultant probability is mostly dominated by the P_{ML} and we have P(w / d) $\cong P_{ML}(w / d)$. When λ is 0.5 the resultant probability is average of the two terms and hence both have equal affect on the resultant probability P(w / d) = 0.5* $P_{ML}(w / d)$ + 0.5*P(w / C). When λ is as high as 0.9, the resultant probability is completely dominated by P(w /C) and we have P(w / d) $\cong P(w / C)$. Comparing this with the results in 2(a) and 2(b) we know that $\lambda = \frac{\mu}{|d| + \mu}$ and (1- λ) is equivalent to $\frac{|d|}{|d| + \mu}$, whenever λ or $\frac{\mu}{|d| + \mu}$ is high P(w / d) $\cong P(w / C)$, and whenever (1- λ) or $\frac{|d|}{|d| + \mu}$ is high P(w / d) $\cong P_{ML}(w / d)$.

3a.

The RSJ model is given by the formula log(O(R=1/ Q,D)),

O(R=1/Q,D) = $\frac{P(R=1/Q,D)}{P(R=0/Q,D)}$ = $\frac{P(D / Q,R=1)P(Q /R=1)P(R=1)}{P(D/Q,R=0)P(Q/R=0)P(R=0)}$, we are ignoring the terms

$P(Q /R = 1), P(R = 1), P(Q/R = 0), P(R = 0)$, then we will have

O(R=1/Q,D) α $\frac{P(D / Q,R=1)}{P(D/Q,R=0)}$

Let D consists of words (w1,w2,w3……..,wd) which are independent of each other. Since words are independent we have

P(D / Q,R=1) = P(w1 / Q,R=1)P(w2 / Q,R=1)……..P(Wd / Q,R=1)

We can write it as a product of words present in the vocabulary and we can iterate over all the words in the vocabulary and then if the word which is present in the vocabulary is not present in the document we can give it a power zero.

P(D / Q,R=1) = $\prod_{w \in |V|}$ P(w / Q, R = 1)^{d_i} where d_i says whether w belongs to document D or not.

We can write the power term as the frequency of the word w in the Document and do this for all the words in the document and hence we get.

P(D / Q,R=1) = $\prod_{w \in |V|}$ P(w / Q, R = 1)$^{c(w,D)}$, where c(w,D) is the frequency of the word in document D.

In a similar way we can show that

P(D / Q,R=0) = $\prod_{w \in |V|}$ P(w / Q, R = 0)$^{c(w,D)}$

Now writing down both the terms in the equation above, we get

O(R=1/Q,D) α $\frac{P(D\ /\ Q,R=1)}{P(D/Q,R=0)}$ = $\frac{\prod_{w \in |V|} \text{P(w / Q,R=1)}^{c(w,D)}}{\prod_{w \in |V|} \text{P(w / Q,R=0)}^{c(w,D)}}$ = $\prod_{w \in |V|} (\frac{\text{P(w / Q,R=1)}}{\text{P(w / Q,R=0)}})^{c(w,D)}$

Now we have O(R=1/Q,D) taking log of it to get the RSJ model we get.

Log(O(R=1/Q,D)) = log($\prod_{w \in |V|} (\frac{\text{P(w / Q,R=1)}}{\text{P(w / Q,R=0)}})^{c(w,D)}$) , we know that log(ab) = log(a) + log(b) and

log(a^b) $= blog(a)$ applying these two to the equation we get

Log(O(R=1/Q,D)) = $\sum_{w \in |V|} (c(w,D)\ log(\frac{\text{P(w / Q,R=1)}}{\text{P(w / Q,R=0)}})$

Given Score(Q,D) = Log(O(R=1/Q,D)) and hence we can say that,

Score(Q , D) = $\sum_{w \in |V|} (c(w,D)\ log(\frac{\text{P(w / Q,R=1)}}{\text{P(w / Q,R=0)}})$

We require 2*|V| parameters in a retrieval model. Where |V| is the Document length as well since we are considering only one document D. if |V| has more words, we just require those words which are present in document because c(w,D) will be zero if word is not present in a document. If we know 2*|V| -1 parameters, we can find the 2*|V|th parameter.
We know that P(w) = P(w / Q, R = 1) + P(w / Q, R = 0) so if we know 2 of the three terms present there we will have the parameters for the retrieval model. To know the two parameters we require 2*|V| parameters in total.

3b.

P(w / Q,R=0) = The Probability of word w present in an irrelevant document.

Given that the no of irrelevant documents is a collection of N documents C = {D1,D2.....Dn) and hence

P(w / Q,R=0) = $\frac{frequency\ of\ w\ in\ the\ collection\ C}{frequency\ of\ all\ the\ words\ in\ the\ collection\ C}$ = $\frac{c(w,C)}{\sum_{i=1}^{N} |D_i|}$

Maximum likelihood estimate is given by

P(w / Q,R=0) = $\frac{c(w,C)}{\sum_{i=1}^{N}|D_i|}$

3c.

P(w / Q,R=1) = The Probability of word w present in a relevant document.

Given that the no of relevant documents is just the query, and hence no of relevant documents is 1.

P(w / Q,R=1) = $\frac{frequency\ of\ w\ in\ a\ relavant\ document}{length\ of\ the\ relavant\ document}$ = $\frac{c(w,D)}{|D|}$

Since we are considering Query is the only relevant document, we will have

P(w / Q,R=1) = $\frac{c(w,Q)}{|Q|}$

3d.

We know that Jelinek-Mercer smoothing is given by the formula

P(w / d) = (1 - λ) $P_{ML}(w / d)$ + λ P(w / C) , using it we get

P(w / Q,R=1) = (1 - λ) P_{ML}(w / Q, R = 1) + λ P_{LM}(w / Q, R = 1) LM is background model.

P(w / Q,R=1) = (1 - λ) $\frac{c(w,Q)}{|Q|}$ + λ P_{LM}(w / Q, R = 1)

3e.

Score(Q , D) = $\sum_{w \in |V|} (c(w,D)\ log(\frac{\text{P(w / Q,R=1)}}{\text{P(w / Q,R=0)}})$

From 3b and 3d we have

P(w / Q,R=0) = $\frac{c(w,C)}{\sum_{i=1}^{N}|D_i|}$, P(w / Q,R=1) = (1 - λ) $\frac{c(w,Q)}{|Q|}$ + λ P_{LM}(w / Q, R = 1),
substituting it we get.

Score(Q , D) = $\sum_{w \in |V|} (c(w,D)\ log(\frac{(1 - \lambda)\frac{c(w,Q)}{|Q|} + \lambda\ P_{LM}(\text{w / Q,R=1})}{\frac{c(w,C)}{\sum_{i=1}^{N}|D_i|}}))$

Score(Q , D) = $\sum_{w \in |V|} (c(w,D)\ log(\frac{((1 - \lambda)\frac{c(w,Q)}{|Q|} + \lambda\ P_{LM}(\text{w / Q,R=1}))}{\frac{c(w,C)}{\sum_{i=1}^{N}|D_i|}}))$

The retrieval function can capture the TF(which is the green part c(w,D)). It can also capture the relative term frequency which is inside the log. the retrieval function cannot capture the IDF and Document length Normalization.