Individual Word Probability Models

Le Quan Ha, Darryl W. Stewart, Ji Ming, F. Jack Smith

School of Electronics, Electrical Engineering and Computer Science,
Queen's University Belfast
Belfast BT7 1NN, Northern Ireland
lequanha@lequanha.com

Abstract

It is shown that the enormous improvement in the size of disk storage space in recent years can be used to build individual word-domain statistical language models, one for each significant word of a language. Each of these word-domain language models is a precise domain model for the relevant significant word and when combined appropriately they provide a highly specific domain language model for the language following a cache, even a short cache. Our individual word probability models have been constructed and tested on the Wall Street Journal Corpus of 40 million words. Improvements in perplexity, between 31.98% and 33.90%, over a base-line tri-gram model have been obtained in tests.

1. Introduction

A human is able to work out the precise domain of a spoken sentence after hearing only a few words. The clear identification of this domain then makes it possible for a human to anticipate the following words and combination of words and thus recognize speech even in a very noisy environment. This anticipation still cannot be replicated by Statistical language models.

Statistical language models have been improving slowly over the last 20 years due to added complexity and larger training corpora, and using the greatly improved processing power of digital computers. However, they have not been using effectively the enormous increase in the availability of disk storage space, which makes it possible today for a student to buy a 300 Gigabyte disk for games and music. This paper suggests one way in which this huge improvement in technology can be used to bring a significant improvement in language modeling, and take a step towards the building of a fairly precise domain model after only a few words of a text

The word-domain language model is based on a simple idea: it extends the idea of cache models (Kuhn and De Mori, 1990) and trigger models (Lau, Rosenfeld and Roukos, 1993) by triggering a separate *n*-gram language model for each content word in a cache and combining them to produce a combined model.

This is done as follows. A training corpus for each significant word is formed from the amalgamation of the text fragments in which that word appears, taken from a large global training corpus. In this paper the text fragments are the sentences in which the significant words occur. Experiments have shown that larger fragments are not needed. A significant word is any word that significantly contributes to the context of the text. We define this as any word which is not a stop word, i.e. not articles, conjunctions or prepositions and not some of the most frequently used words in the language such as "will" and not common adverbs and adjectives such as "now", "very", "some", etc. So we assume that all other words are significant and a corpus is built for each. A statistical language model is then calculated from this corpus, i.e. from all of the sentences containing the word. So it should be able to represent the domain of that word. There are a very large number of individual word language models. The only requirement is enormous quantities of disk space which are now available even on a PC.

2. The Language Models

It was found in experiments that we needed to combine the global language model with the individual word-domain models to obtain good results. (This may be due to the limited size of the

global corpus in our tests, 40 million tokens.) So we first build a language model for the whole global corpus. Frequencies of words and phrases are derived from the corpus and the conditional probability of a word given a sequence of preceding words is calculated. The individual conditional probabilities are approximated by the maximum likelihoods

$$P_{ML}\left(w_i \middle| w_1^{i-1}\right) = \frac{f\left(w_1^i\right)}{f\left(w_1^{i-1}\right)} = \frac{f\left(w_1 \ldots w_{i-1} w_i\right)}{f\left(w_1 \ldots w_{i-1}\right)} \qquad (1)$$

where $f\left(w_1^n\right)$ is the frequency of the phrase $w_1^n = w_1 \ldots w_{n-1} w_n$ in the text. These probabilities are smoothed by one of a number of well known methods such as Turing-Good estimation (Good, 1953), the Katz back-off method (Katz, 1987) or others. Although any of these could be used in our experiment to demonstrate the principle of our multiple word-domain model, it was convenient to use the empirical weighted average (WA) linear interpolation *n*-gram model (O'Boyle, Owens and Smith, 1994) because of its simplicity. It gives results comparable to the Katz back-off method but is much quicker to use. The weighted average probability of a word w given the preceding words $w_1 \ldots w_{n-1} w_n$ is

$$P_{WA}\left(w \middle| w_1^m\right) = \frac{\mu_0 P_{ML}(w) + \sum_{i=1}^{m} \mu_i P_{ML}\left(w \middle| w_{m+1-i}^m\right)}{\sum_{i=0}^{m} \mu_i} \qquad (2)$$

where the weighted functions (in the simplest case) are given by

$$\mu_0 = Ln(T) \quad and \quad \mu_i = Ln\left(f\left(w_{m+1-i}^m\right)\right).2^i \qquad (3)$$

T is the number of tokens in the corpus and $f\left(w_{m+1-i}^m\right)$ is the frequency of the sentence $w_{m+1-i} \ldots w_m$ in the text.

The unigram maximum likelihood probability of a word is

$$P_{ML}(w) = \frac{f(w)}{T} \qquad (4)$$

The language model defined by equation (2) and (4) is called the global language model when trained on the global corpus. Following the creation of the global model comes the creation of a language model for each significant word, which is formed in the same manner as the global language model.

3. Probability Models

We need to combine the probabilities obtained from each word domain language model and from the global language model, in order to obtain a combined probability for a word given a sequence of words. One simple way for doing this is an arithmetic combination of the global language model and the word language models in a linear interpolated expression as follows

$$P\left(w \middle| w_1^n\right) = \lambda_G P_{Global}\left(w \middle| w_1^n\right) + \sum_{i=1}^{m} \lambda_i P_i\left(w \middle| w_1^n\right) \qquad (5)$$

where $\lambda_G + \sum_{i=1}^{m} \lambda_i = 1$, where $P_{Global}\left(w \middle| w_1^n\right)$ is the conditional probability of the word w following a phrase $w_1 \ldots w_{n-1}^1 w_n$ in the global language model, and P_i is the conditional probability in the word language model for the significant word w_i, λ_i is the correspondent weight and m is the number of word models that are included.

Ideally the λ_i parameters would be optimised using a held-out training corpus; however this is not practical as we do not know which combination of words w_i will arise in the cache. So a simpler approach is needed.

3.1. Linear Interpolation

A simple way of choosing the λ values is to give the same weight to all the word language models but a different weight to the global language model, and put a restriction on the number of word language models to be included. This weighted model is defined as

$$P\left(w\middle|w_1^n\right)=\lambda.P_{Global}\left(w\middle|w_1^n\right)+\frac{(1-\lambda)}{m}\left[\sum_{i=1}^{m}P_i\left(w\middle|w_1^n\right)\right] \qquad (6)$$

and λ and m are parameters which are chosen to optimise the model.

3.2. Exponential Decay Model

A method was developed based on an exponential decay of the word model probabilities with distance since a word appearing several words previously will generally be less relevant than more recent words. Given a sequence of words, for example, "We had happy times in Spain…" in Table 1.

Table 1: An explanation of the exponential decay model

We	Had	**Happy**	**Times**	In	**Spain**
5	4	3	2	1	

where 5, 4, 3, 2, 1 represent the distance of the word from the word *Spain*. The words *Happy* and *Times* are significant words for which we have individual word language models. The exponential decay model for the word *w*, where in this case *w* represents the significant word *Spain*, is as follows

$$P\left(w\middle|w_1^n\right)=\frac{P_{Global}\left(w\middle|w_1^n\right)+P_{Happy}\left(w\middle|w_1^n\right).\exp(-3/d)+P_{Times}\left(w\middle|w_1^n\right).\exp(-2/d)}{1+\exp(-3/d)+\exp(-2/d)} \qquad (7)$$

where $P_{Global}\left(w\middle|w_1^n\right)$ is the conditional probability of the word *w* following a phrase $w_1w_2\ldots w_n$ in the global language model, $P_{Happy}\left(w\middle|w_1^n\right)$ is the conditional probability of the word *w* following a phrase $w_1\ldots w_{n-1}w_n$ in the word language model for the significant word *Happy*. The same definition applies for the word model *Times*. *d* is the exponential decay distance with *d* = 5, 10, 15, etc. A cache or cut-off is introduced in the model

$$\text{if } l \geq cache \Rightarrow \text{replace } \exp(-l/d) \text{ by } 0$$

where l is the distance from the significant word to the target word and *d* is the decay distance.

3.3. Weighted Models

In the two methods above, the weights for the word language models were independent of the size of the word training corpora or the global training corpus. So we introduce new weights to these models which depend on the size of the training corpora. These weights used are functions of the size of the word training corpora, that is, the number of tokens of the training corpora T_i. Some examples of the weights can be seen in Table 2.

Table 2: Some of the weights used in this model

Weights
$\text{Ln}(1+\text{Ln}T_i)$
$\text{Sqrt}(\text{Ln}T_i)$
$\text{Ln}T_i$
$\text{Sqrt}(T_i)$
$T_i/\text{Ln}T_i$
T_i
$T_i\text{Ln}T_i$

3.3.1. Weighted Probability Model

The weighted probability model is based on the idea that the weight given to a word language model should depend on the size of the training corpora. It is described in the following equation

$$P\left(w\middle|w_1^n\right)=\frac{\beta_{Global}.P_{Global}\left(w\middle|w_1^n\right)+\sum_{i=1}^{m}\beta_i.P_i\left(w\middle|w_1^n\right)}{\beta_{Global}+\sum_{i=1}^{m}\beta_i} \quad (8)$$

where β_{Global} is the weight for the global language model and β_i is the weight for the word model for the word w_i. We give more weight to those word models with small training corpus, as they represent models for the less frequent words which therefore have the most information. The weights used are functions of the size of the word training corpora, that is, of the number of tokens of the training corpora T_i. The optimum functions were found by experimenting with the weights also in the Table 2.

3.3.2. Weighted Exponential Model

The weighted exponential language model is a combination of the weighted probability model (section 3.2) and the exponential decay model (section 3.3.1). Each language model has two functions, one is the exponential decay in terms of the distance from the significant word and the second function is a weight that depends on the size of the word language model training corpora. Given the same example as the one in section 3.2 based on the sentence "We had happy times in Spain", we define the weighted exponential decay model for the word w, where in this case w represents the significant word *Spain*, as follows

$$P\left(w\middle|w_1^n\right)=\frac{\beta_{Global}P_{Global}\left(w\middle|w_1^n\right)+\beta_{Happy}P_{Happy}\left(w\middle|w_1^n\right).\exp(-3/d)+\beta_{Times}P_{Times}\left(w\middle|w_1^n\right).\exp(-2/d)}{\beta_{Global}+\beta_{Happy}\exp(-3/d)+\beta_{Times}\exp(-2/d)} \quad (9)$$

where $P_{Global}\left(w\middle|w_1^n\right)$ is the conditional probability of the word w following a phrase $w_1w_2\ldots w_n$ in the global language model, $P_{Happy}\left(w\middle|w_1^n\right)$ is the conditional probability of the word w following a phrase $w_1\ldots w_{n-1}w_n$ in the word language model for the significant word *Happy*. The same definition applies for the word model *Times*. d is the exponential decay distance with d = 5, 10, 15, etc. The weight β_{Global} is the weight for the global language model and β_i is the weight for the word model for the word w_i. The values of the functions β are those used before (Table 2.)

3.3.3. Linear Interpolation Exponential Model with weights

Finally, we decided to try another model based on a combination of all of the previous methods. It is based on the idea that perhaps the global language model should be weighted in a different way from the word language models. This is equivalent to a combination of all the methods seen previously in one model that we called linear interpolation exponential model with weights. The probability of a word given the previous words is

$$P\left(w\middle|w_1^n\right)=\lambda P_{Global}\left(w\middle|w_1^n\right)+(1-\lambda)\frac{\beta_{Happy}P_{Happy}\left(w\middle|w_1^n\right)\exp(-3/d)+\beta_{Times}P_{Times}\left(w\middle|w_1^n\right)\exp(-2/d)}{\beta_{Happy}\exp(-3/d)+\beta_{Times}\exp(-2/d)} \quad (10)$$

where the β_i functions are the same as in previous models and λ is a separate parameter to be determined.

4. Methods of testing

Perplexity is a well known measure of the performance of a language model (Jelinek, Mercer and Bahl, 1983). We calculate the perplexity of each sentence, w_1^n, using the formula:

$$P(w_1^n)=P(w_1)P(w_2\mid w_1)\cdots P(w_n\mid w_1^{n-1}) \quad (11)$$

and the perplexity by

$$PP(w_1^n) = \exp\left(-\frac{1}{m}\sum_{i=1}^{m} Ln(P(w_i \mid w_1 w_2 \ldots w_{i-1})\right) \quad (12)$$

Our method of calculating the constituent probabilities on the right-hand side of equation (12) is employing the word domain language models with *a priori* method as reported in earlier papers on this method by Sicilia-Garcia, Ming and Smith (2001, 2002).

In the *a priori* method at the beginning of the sentence, since we do not know which significant words are going to appear in the sentence, we use the global language model and possibly individual word models from earlier sentences (i.e. from the cache). We then add in a word language model for each significant word after it appears in the sentence. Thus in the sentence

"The cat sat on the mat"

neglecting previous sentences, the first two words are modeled using the global language model, the probability P(sat| the cat) is calculated using the global model combined with the word for "cat", and the last three words are modeled using the global model combined with the word models for "cat" and "sat".

5. Corpus

The methods described above were compared in some experiments using the Wall Street Journal (WSJ) corpus (Paul and Baker, 1992). Previous research by Sicilia-Garcia et al. (2001, 2002) compared how the individual word probability models depend on the size of the training corpus for two subsets of the WSJ of 16 million (1988) and 6 million words (1989) approximately. The well-known WSJ test file (Paul et al., 1992) contains 584 paragraphs, 1,869 sentences, 34,781 tokens and 3,677 words types. Now we are developing these models for the combined WSJ corpus of 40 million words and presenting here more in details.

The results reveal a lower perplexity for the larger 40 million word corpus (as we would expect). These results are then compared back to Sicilia-Garcia's work in this paper.

6. Results

We gradually present the results for each probability models starting from the Linear Interpolation Model.

6.1. Linear Interpolation Probability Model of the combined WSJ of 40 million words

The perplexity results are shown in Table 3 and the improvement results are shown in Table 4 (WM is the maximum number of word language models included into the new combination model.)

Table 3: Perplexity results of the Linear Interpolation Probability Model

***n*-gram**	**WM**	**λ=0.1**	**λ=0.2**	**λ=0.3**	**λ=0.4**	**λ=0.5**	**λ=0.6**	**λ=0.7**	**λ=0.8**	**λ=0.9**	**λ=1.0**
3	10	72.286	69.500	68.045	67.280	67.008	67.169	67.777	68.934	70.932	75.36
	11	72.342	69.539	68.073	67.298	67.018	67.172	67.775	68.927	70.923	75.36
	12	72.378	69.567	68.093	67.312	67.026	67.175	67.773	68.921	70.915	75.36
	13	72.428	69.611	68.132	67.345	67.054	67.198	67.792	68.936	70.925	75.36
	14	72.433	69.607	68.122	67.330	67.035	67.176	67.767	68.910	70.902	75.36
	15	72.426	69.596	68.108	67.315	67.019	67.16	67.751	68.897	70.894	75.36
	16	72.345	69.519	68.033	67.241	**66.947**	67.089	67.682	68.831	70.836	75.36
	17	72.374	69.542	68.052	67.256	66.959	67.097	67.688	68.833	70.835	75.36
	18	72.408	69.568	68.072	67.271	66.968	67.102	67.689	68.831	70.83	75.36
	19	72.433	69.586	68.084	67.279	66.972	67.103	67.686	68.825	70.823	75.36

	20	72.45	69.601	68.096	67.287	66.978	67.106	67.687	68.824	70.82	75.36
5	10	60.521	57.435	55.582	54.359	53.569	53.138	53.051	53.360	54.244	56.845
	20	60.748	57.54	55.599	54.31	53.47	53	52.888	53.188	54.091	56.845
	21	60.747	57.535	55.592	54.301	53.46	52.99	52.878	53.178	54.084	56.845
	22	60.755	57.541	55.596	54.304	53.462	52.992	52.88	53.179	54.085	56.845
	23	60.752	57.538	55.592	54.299	53.457	52.986	**52.874**	53.175	54.082	56.845
	24	60.766	57.548	55.599	54.305	53.461	52.989	52.875	53.175	54.081	56.845
	25	60.776	57.555	55.605	54.309	53.464	52.991	52.877	53.176	54.082	56.845
7	10	59.362	56.238	54.340	53.07	52.234	51.755	51.619	51.878	52.713	55.289
	20	59.58	56.322	54.33	52.991	52.103	51.587	51.429	51.683	52.544	55.289
	21	59.579	56.317	54.322	52.982	52.093	51.577	51.419	51.674	52.537	55.289
	22	59.586	56.322	54.326	52.984	52.095	51.578	51.42	51.675	52.538	55.289
	23	59.582	56.316	54.319	52.977	52.088	51.571	**51.413**	51.668	52.533	55.289
	24	59.596	56.327	54.327	52.983	52.091	51.573	51.414	51.668	52.532	55.289
	25	59.605	56.334	54.332	52.987	52.094	51.575	51.415	51.669	52.532	55.289
9	10	59.258	56.125	54.219	52.943	52.101	51.618	51.481	51.741	52.582	55.183
	20	59.458	56.191	54.191	52.847	51.955	51.437	51.279	51.535	52.404	55.183
	21	59.456	56.185	54.184	52.838	51.945	51.427	51.269	51.526	52.397	55.183
	22	59.463	56.19	54.187	52.84	51.946	51.427	51.269	51.527	52.398	55.183
	23	59.458	56.184	54.18	52.833	51.939	51.42	**51.262**	51.52	52.393	55.183
	24	59.472	56.195	54.188	52.838	51.943	51.422	51.263	51.52	52.392	55.183
	25	59.482	56.202	54.193	52.842	51.946	51.424	51.264	51.521	52.392	55.183

Table 4: Perplexity improvement of the Linear Interpolation Probability Model

***n*-gram**	**WM**	**λ=0.1**	**λ=0.2**	**λ=0.3**	**λ=0.4**	**λ=0.5**	**λ=0.6**	**λ=0.7**	**λ=0.8**	**λ=0.9**	**λ=1.0**
3	10	4.079%	7.776%	9.706%	10.722%	11.082%	10.869%	10.062%	8.527%	5.876%	0.000%
	11	4.005%	7.724%	9.670%	10.698%	11.069%	10.865%	10.065%	8.536%	5.887%	0.000%
	12	3.957%	7.687%	9.643%	10.680%	11.059%	10.861%	10.068%	8.544%	5.898%	0.000%
	13	3.891%	7.629%	9.592%	10.636%	11.021%	10.830%	10.043%	8.525%	5.884%	0.000%
	14	3.883%	7.633%	9.605%	10.656%	11.047%	10.860%	10.076%	8.559%	5.915%	0.000%
	15	3.893%	7.648%	9.623%	10.676%	11.068%	10.882%	10.096%	8.577%	5.927%	0.000%
	16	4.001%	7.751%	9.723%	10.774%	**11.164%**	10.976%	10.188%	8.664%	6.003%	0.000%
	17	3.962%	7.720%	9.698%	10.753%	11.148%	10.964%	10.181%	8.661%	6.005%	0.000%
	18	3.917%	7.685%	9.671%	10.734%	11.136%	10.958%	10.180%	8.664%	6.011%	0.000%
	19	3.884%	7.661%	9.654%	10.723%	11.130%	10.957%	10.183%	8.671%	6.020%	0.000%
	20	3.861%	7.642%	9.639%	10.712%	11.123%	10.953%	10.182%	8.674%	6.025%	0.000%
5	10	19.691%	23.785%	26.245%	27.868%	28.915%	29.488%	29.603%	29.193%	28.020%	24.568%
	20	19.390%	23.646%	26.222%	27.933%	29.048%	29.671%	29.819%	29.422%	28.223%	24.568%
	21	19.392%	23.653%	26.232%	27.945%	29.061%	29.684%	29.832%	29.434%	28.232%	24.568%
	22	19.381%	23.645%	26.226%	27.941%	29.058%	29.682%	29.831%	29.433%	28.231%	24.568%
	23	19.384%	23.650%	26.232%	27.947%	29.065%	29.689%	**29.837%**	29.439%	28.235%	24.568%
	24	19.366%	23.636%	26.222%	27.940%	29.060%	29.686%	29.836%	29.439%	28.236%	24.568%
	25	19.352%	23.626%	26.214%	27.933%	29.055%	29.683%	29.834%	29.438%	28.236%	24.568%
7	10	21.228%	25.374%	27.893%	29.578%	30.688%	31.324%	31.503%	31.159%	30.051%	26.634%

	20	20.939%	25.263%	27.906%	29.683%	30.861%	31.546%	31.755%	31.418%	30.276%	26.634%
	21	20.941%	25.270%	27.916%	29.695%	30.874%	31.559%	31.769%	31.430%	30.285%	26.634%
	22	20.931%	25.263%	27.912%	29.692%	30.872%	31.558%	31.767%	31.429%	30.284%	26.634%
	23	20.937%	25.270%	27.920%	29.701%	30.882%	31.568%	**31.777%**	31.438%	30.290%	26.634%
	24	20.919%	25.257%	27.910%	29.694%	30.877%	31.565%	31.776%	31.438%	30.292%	26.634%
	25	20.906%	25.247%	27.903%	29.688%	30.873%	31.562%	31.774%	31.438%	30.292%	26.634%
9	10	21.366%	25.524%	28.054%	29.747%	30.864%	31.505%	31.687%	31.342%	30.226%	26.774%
	20	21.102%	25.437%	28.090%	29.874%	31.057%	31.745%	31.955%	31.615%	30.462%	26.774%
	21	21.104%	25.444%	28.100%	29.886%	31.070%	31.759%	31.968%	31.627%	30.471%	26.774%
	22	21.095%	25.438%	28.096%	29.884%	31.069%	31.758%	31.968%	31.626%	30.470%	26.774%
	23	21.101%	25.445%	28.105%	29.893%	31.078%	31.768%	**31.977%**	31.635%	30.476%	26.774%
	24	21.082%	25.432%	28.095%	29.886%	31.074%	31.765%	31.976%	31.635%	30.477%	26.774%
	25	21.070%	25.422%	28.088%	29.880%	31.070%	31.762%	31.974%	31.634%	30.477%	26.774%

Here and after are our overall results combined to smaller corpus-sizes of Sicilia-Garcia et al. (2001, 2002) in Table 5. Their WSJ88 and WSJ89 results were not for 7-grams.

Table 5: Resume for the Linear Interpolation Probability Model

		Sentence/WSJ			**Sentence/WSJ88**			**Sentence/WSJ89**		
		Perplex-ity	**Improve-ment**	***(λ,WM)***	**Perplex-ity**	**Improve-ment**	***(λ,WM)***	**Perplex-ity**	**Improve-ment**	***(λ,WM)***
***n*-gram**	3	66.95	11.16%	*(0.5, 16)*	78.51	9.27%	*(0.6, 21)*	99.16	7.48%	*(0.7, 23)*
	5	52.87	29.84%	*(0.7, 23)*	65.61	24.18%	*(0.7, 21)*	88.49	17.44%	*(0.7, 27)*
	7	51.41	31.78%	*(0.7, 23)*	-	-	-	-	-	-
	9	51.26	31.98%	*(0.7, 23)*	64.54	25.41%	*(0.7, 22)*	88.12	17.78%	*(0.7, 27)*

From Table 5, we can extract the following information

1) An improvement of 32% is obtained with respect to the tri-gram global model, using the combined WSJ, 23 word models.
2) The optimum value of λ is still between 0.5 and 0.7 as previous authors (Sicilia-Garcia et al., 2001, 2002)'s observation. The word language models have a smaller weight than the global language model. It seems that as the size of the training corpus gets bigger, the importance of the word models increases.

6.2. Exponential Decay Model of the combined WSJ of 40 million words

The perplexity results are shown in Table 6 and the improvement results are shown in Table 7.

Table 6: Perplexity results of the Exponential Decay Model

		tri-gram			**5-gram**			**9-gram**		
		d			***d***			***d***		
		8	**9**	**10**	**5**	**6**	**7**	**5**	**6**	**7**
Cache	**25**	63.062264	63.110652	63.186005	51.368222	51.340501	51.411274	50.132961	50.122189	50.204252
	30	62.903743	62.939731	63.008784	51.304749	51.247012	51.295403	50.065523	50.022956	50.081043
	35	62.817057	62.841173	62.902443	51.280116	51.204316	51.236979	50.037772	49.976669	50.018779
	40	62.774785	62.789876	62.844587	51.273242	51.188954	51.212522	50.030612	49.960673	49.993313
	45	62.750838	62.758826	62.807378	51.270169	51.181117	51.198687	50.027382	49.952408	49.978654
	50	62.741415	62.745713	62.791080	51.269365	51.178390	51.193048	50.024871	49.946815	49.969658
	55	62.737315	62.739326	62.782310	51.269190	51.177666	51.191214	50.024683	49.946037	49.967702
	60	62.736074	62.737222	62.779261	51.269148	51.177459	51.190620	50.024639	49.945825	49.967099

	65	62.734829	62.734789	62.775235	51.269123	51.177319	51.190148	50.024610	49.945673	49.966598
	70	62.734798	62.734740	62.775172	51.269122	51.177315	51.190137	50.024609	**49.945668**	49.966585
	75	62.734794	**62.734737**	62.775177	51.269122	**51.177314**	51.190137	50.024609	49.945668	49.966584
	80	62.734796	62.734743	62.775189	51.269122	51.177314	51.190137	50.024609	49.945668	49.966585
	85	62.734797	62.734746	62.775196	51.269122	51.177314	51.190138	50.024609	49.945668	49.966585
	90	62.734798	62.734749	62.775204	51.269122	51.177315	51.190138	50.024609	49.945668	49.966585
	95	62.734798	62.734751	62.775208	51.269122	51.177315	51.190138	50.024609	49.945668	49.966585
	100	62.734798	62.734750	62.775206	51.269122	51.177315	51.190138	50.024609	49.945668	49.966585

Table 7: Perplexity improvements of the Exponential Decay Model

		tri-gram			**5-gram**			**9-gram**		
		d			***d***			***d***		
		8	**9**	**10**	**5**	**6**	**7**	**5**	**6**	**7**
Cache	**25**	16.318640%	16.254431%	16.154440%	31.836214%	31.872998%	31.779085%	33.475361%	33.489655%	33.380760%
	30	16.528991%	16.481237%	16.389606%	31.920440%	31.997055%	31.932842%	33.564848%	33.621333%	33.544254%
	35	16.644021%	16.612020%	16.530717%	31.953127%	32.053711%	32.010368%	33.601673%	33.682755%	33.626876%
	40	16.700114%	16.680089%	16.607489%	31.962249%	32.074096%	32.042822%	33.611174%	33.703981%	33.660669%
	45	16.731891%	16.721291%	16.656864%	31.966327%	32.084495%	32.061181%	33.615460%	33.714948%	33.680121%
	50	16.744395%	16.738692%	16.678491%	31.967393%	32.088114%	32.068663%	33.618792%	33.722370%	33.692058%
	55	16.749835%	16.747167%	16.690129%	31.967626%	32.089075%	32.071097%	33.619042%	33.723402%	33.694653%
	60	16.751482%	16.749959%	16.694175%	31.967681%	32.089349%	32.071885%	33.619100%	33.723683%	33.695454%
	65	16.753134%	16.753187%	16.699517%	31.967715%	32.089535%	32.072511%	33.619139%	33.723885%	33.696118%
	70	16.753175%	16.753252%	16.699601%	31.967716%	32.089540%	32.072526%	33.619140%	**33.723892%**	33.696136%
	75	16.753181%	**16.753256%**	16.699594%	31.967716%	**32.089542%**	32.072526%	33.619140%	33.723892%	33.696137%
	80	16.753178%	16.753248%	16.699578%	31.967716%	32.089542%	32.072526%	33.619140%	33.723892%	33.696136%
	85	16.753177%	16.753244%	16.699569%	31.967716%	32.089542%	32.072525%	33.619140%	33.723892%	33.696136%
	90	16.753175%	16.753240%	16.699558%	31.967716%	32.089540%	32.072525%	33.619140%	33.723892%	33.696136%
	95	16.753175%	16.753238%	16.699553%	31.967716%	32.089540%	32.072525%	33.619140%	33.723892%	33.696136%
	100	16.753175%	16.753239%	16.699555%	31.967716%	32.089540%	32.072525%	33.619140%	33.723892%	33.696136%

Our total results combined to previous results (Sicilia-Garcia et al., 2001, 2002) in Table 8 (their WSJ88 and WSJ89 results were not for 7-grams.)

Table 8: Resume for the Exponential Decay Model

		Sentence/WSJ			**Sentence/WSJ88**			**Sentence/WSJ89**		
		Perplex-ity	**Improve-Ment**	***(d,Cache)***	**Perplex-ity**	**Improve-ment**	***(d,Cache)***	**Perplex-ity**	**Improve-ment**	***(d,Cache)***
***n*-gram**	3	62.73	16.75%	*(9, 75)*	74.91	13.43%	*(7, 50)*	97.32	9.20%	*(5, 50)*
	5	51.18	32.09%	*(6, 75)*	64.42	25.55%	*(5, 40)*	88.54	17.39%	*(4, 40)*
	7	50.03	33.61%	*(6, 75)*	-	-	-	-	-	-
	9	49.95	33.72%	*(6,70)*	63.70	26.39%	*(5, 50)*	88.46	17.47%	*(5, 40)*

From these results, we outline the conclusions for the exponential decay model.

1. An improvement up to 34% is obtained with respect to the tri-gram global language model with a decay distance of 6 and a cache of 70 words. This is a small improvement over the linear interpolation model (32%)
2. The exponential decay model performs better with bigger training corpora probably because the information available is bigger

6.3. Weighted Model of the combined WSJ of 40 million words

6.3.1. Weighted Probability Model

The perplexity results are shown in Table 9 and the improvement results are shown in Table 10.

Table 9: Perplexity results of the Weighted Probability Model

n-gram	WM	T_iLnT_i	T_i	T_i/LnT_i	SqrtT_i	LnT_i	Sqrt(LnT_i)	Ln(1+LnT_i)	1/Ln(1+LnT_i)
3	**5**	73.084	72.713	72.268	67.599	**64.882**	65.019	65.070	65.410
	6	72.827	72.416	71.929	67.149	64.910	65.074	65.133	65.503
	7	72.610	72.172	71.655	66.838	64.950	65.131	65.194	65.582
	8	72.410	71.947	71.405	66.578	64.969	65.160	65.226	65.625
	9	72.256	71.775	71.215	66.428	65.093	65.297	65.366	65.782
	10	72.144	71.652	71.080	66.353	65.211	65.422	65.493	65.917
	15	71.759	71.230	70.625	66.108	65.459	65.690	65.766	66.216
	20	71.572	71.028	70.410	66.013	65.529	65.768	65.846	66.306
	25	71.513	70.967	70.350	66.018	65.580	65.820	65.899	66.363
	30	71.506	70.961	70.346	66.039	65.617	65.859	65.939	66.405
5	**10**	55.159	54.881	54.562	52.382	54.031	54.410	54.530	55.157
	15	54.951	54.653	54.316	52.377	54.438	54.832	54.956	55.602
	16	54.910	54.606	54.264	**52.338**	54.418	54.812	54.936	55.582
	17	54.895	54.590	54.248	52.358	54.470	54.864	54.989	55.637
	18	54.877	54.571	54.229	52.373	54.519	54.914	55.039	55.689
	19	54.865	54.559	54.217	52.383	54.554	54.952	55.077	55.730
	20	54.855	54.549	54.208	52.399	54.587	54.986	55.111	55.765
	25	54.828	54.522	54.182	52.433	54.651	55.050	55.177	55.833
	30	54.830	54.525	54.188	52.465	54.697	55.098	55.225	55.883
7	**10**	53.637	53.367	53.057	51.023	52.927	53.326	53.453	54.106
	15	53.429	53.138	52.811	51.021	53.339	53.753	53.883	54.556
	16	53.389	53.092	52.760	**50.984**	53.323	53.737	53.867	54.540
	17	53.375	53.078	52.746	51.006	53.377	53.792	53.923	54.598
	18	53.357	53.059	52.727	51.022	53.428	53.844	53.975	54.651
	19	53.345	53.047	52.714	51.033	53.464	53.883	54.014	54.693
	20	53.334	53.035	52.704	51.048	53.498	53.917	54.049	54.730
	25	53.307	53.008	52.679	51.083	53.562	53.983	54.115	54.798
	30	53.309	53.012	52.684	51.116	53.609	54.031	54.164	54.849
9	**10**	53.518	53.246	52.935	50.900	52.825	53.227	53.355	54.011
	15	53.302	53.009	52.680	50.888	53.228	53.645	53.776	54.453
	16	53.262	52.963	52.629	**50.850**	53.211	53.628	53.759	54.436
	17	53.247	52.948	52.614	50.870	53.263	53.681	53.813	54.492
	18	53.228	52.928	52.594	50.884	53.313	53.732	53.864	54.544
	19	53.214	52.913	52.579	50.893	53.349	53.770	53.902	54.586
	20	53.202	52.901	52.568	50.908	53.382	53.804	53.937	54.622
	25	53.174	52.874	52.542	50.942	53.445	53.869	54.002	54.690
	30	53.176	52.877	52.547	50.974	53.493	53.917	54.051	54.740

Table 10: Perplexity improvements of the Weighted Probability Model

n-gram	WM	T_iLnT_i	T_i	T_i/LnT_i	SqrtT_i	LnT_i	Sqrt(LnT_i)	Ln(1+LnT_i)	1/Ln(1+LnT_i)
3	**5**	3.020%	3.513%	4.103%	10.298%	**13.904%**	13.722%	13.654%	13.203%
	6	3.362%	3.906%	4.553%	10.896%	13.867%	13.649%	13.571%	13.080%
	7	3.649%	4.230%	4.916%	11.308%	13.813%	13.574%	13.490%	12.975%
	8	3.915%	4.529%	5.249%	11.653%	13.788%	13.535%	13.448%	12.919%

	9	4.119%	4.757%	5.501%	11.852%	13.624%	13.354%	13.262%	12.710%
	10	4.268%	4.920%	5.679%	11.952%	13.467%	13.187%	13.093%	12.530%
	15	4.778%	5.481%	6.284%	12.277%	13.139%	12.832%	12.731%	12.133%
	20	5.026%	5.749%	6.568%	12.403%	13.045%	12.729%	12.624%	12.014%
	25	5.105%	5.829%	6.648%	12.396%	12.978%	12.659%	12.554%	11.939%
	30	5.114%	5.837%	6.654%	12.368%	12.928%	12.607%	12.502%	11.883%
5	**10**	26.806%	27.174%	27.598%	30.491%	28.303%	27.800%	27.640%	26.809%
	15	27.082%	27.478%	27.925%	30.497%	27.763%	27.241%	27.075%	26.218%
	16	27.136%	27.540%	27.994%	**30.550%**	27.789%	27.267%	27.102%	26.244%
	17	27.156%	27.560%	28.014%	30.523%	27.721%	27.197%	27.032%	26.172%
	18	27.181%	27.586%	28.040%	30.504%	27.656%	27.131%	26.965%	26.103%
	19	27.196%	27.602%	28.056%	30.490%	27.609%	27.081%	26.914%	26.048%
	20	27.209%	27.615%	28.068%	30.469%	27.565%	27.036%	26.869%	26.002%
	25	27.245%	27.651%	28.102%	30.423%	27.481%	26.950%	26.783%	25.911%
	30	27.243%	27.647%	28.095%	30.380%	27.419%	26.887%	26.719%	25.845%
7	**10**	28.826%	29.184%	29.596%	32.294%	29.768%	29.238%	29.070%	28.204%
	15	29.101%	29.488%	29.922%	32.297%	29.221%	28.672%	28.499%	27.606%
	16	29.155%	29.548%	29.989%	**32.346%**	29.242%	28.693%	28.520%	27.627%
	17	29.173%	29.567%	30.007%	32.317%	29.170%	28.620%	28.446%	27.551%
	18	29.198%	29.593%	30.033%	32.296%	29.103%	28.551%	28.377%	27.480%
	19	29.213%	29.609%	30.050%	32.282%	29.055%	28.500%	28.325%	27.424%
	20	29.228%	29.624%	30.064%	32.261%	29.010%	28.454%	28.278%	27.376%
	25	29.264%	29.660%	30.097%	32.214%	28.925%	28.367%	28.191%	27.285%
	30	29.261%	29.655%	30.090%	32.171%	28.862%	28.303%	28.127%	27.218%
9	**10**	28.984%	29.344%	29.757%	32.457%	29.903%	29.370%	29.200%	28.329%
	15	29.270%	29.659%	30.095%	32.474%	29.369%	28.816%	28.641%	27.743%
	16	29.323%	29.720%	30.163%	**32.524%**	29.391%	28.838%	28.664%	27.765%
	17	29.343%	29.740%	30.183%	32.497%	29.321%	28.767%	28.592%	27.692%
	18	29.369%	29.767%	30.210%	32.478%	29.256%	28.700%	28.524%	27.622%
	19	29.387%	29.786%	30.229%	32.466%	29.208%	28.650%	28.473%	27.567%
	20	29.403%	29.802%	30.245%	32.447%	29.164%	28.604%	28.427%	27.519%
	25	29.440%	29.838%	30.279%	32.401%	29.080%	28.518%	28.341%	27.429%
	30	29.437%	29.834%	30.272%	32.359%	29.017%	28.454%	28.277%	27.362%

Our total results combined to smaller corpus-sizes (Sicilia-Garcia et al., 2001, 2002) in Table 11 (their WSJ88 and WSJ89 results were not for 7-grams.)

Table 11: Resume for the Weighted Probability Model

		Sentence/WSJ			**Sentence/WSJ88**			**Sentence/WSJ89**		
		Perplex-ity	**Improve-ment**	***(WM, Function)***	**Perplex-ity**	**Improve-Ment**	***(WM, Function)***	**Perplex-ity**	**Improve-ment**	***(WM, Function)***
***n*-gram**	3	64.88	13.90%	*(5, LnT_i)*	77.01	11.00%	*(21, $Sqrt(T_i)$)*	97.07	9.43%	*(21, $Sqrt(T_i)$)*
	5	52.34	30.55%	*(16, $Sqrt(T_i)$)*	64.91	24.98%	*(14, $Sqrt(T_i)$)*	87.38	18.47%	*(21, $Sqrt(T_i)$)*
	7	50.98	32.35%	*(16, $Sqrt(T_i)$)*	-	-	-	-	-	-
	9	50.85	32.52%	*(16, $Sqrt(T_i)$)*	63.93	26.11%	*(16, $Sqrt(T_i)$)*	87.04	18.79%	(23, *$Sqrt(T_i)$*)

The following conclusions can be drawn

1) An improvement of up to 33% is obtained with respect to the tri-gram global language model. This optimum corresponds to a combination of up to 16 word language models using the weighted function *$Sqrt(T_i)$*
2) The optimum number of word language models seems to be standard. This indicates that the word language models are good estimators of the probability

6.3.2. *Weighted Exponential Model*

Table 12: Perplexity results of the Weighted Exponential Model

***n*-gram**	***d***	**Cache**	**Sqrt(T_i)**	**LnT_i**	**Sqrt(LnT_i)**	**Ln(1+LnT_i)**	**1/Ln(1+LnT_i)**	**Sqrt(1/LnT_i)**	**1/LnT_i**
3	7	60	69.236593	63.186346	62.955503	62.896228	62.748592	62.730581	62.749806
	7	65	69.236497	63.185928	62.955043	62.895755	62.748057	62.730030	62.749203
	7	70	69.236491	63.185914	62.955028	62.895740	62.748041	62.730014	62.749185
	7	75	69.236488	63.185911	62.955025	62.895737	62.748038	62.730011	62.749183
	7	80	69.236488	63.185911	62.955026	62.895737	62.748039	62.730012	62.749183
	7	85	69.236487	63.185911	62.955026	62.895738	62.748039	62.730012	62.749184
	8	60	68.775078	63.023321	62.841602	62.797405	62.716694	62.712940	62.777760
	8	65	68.774817	63.022272	62.840458	62.796230	62.715376	62.711586	62.776281
	8	70	68.774800	63.022245	62.840429	62.796200	62.715343	62.711552	62.776243
	8	75	68.774792	63.022239	62.840424	62.796195	62.715340	**62.711549**	62.776241
	8	80	68.774790	63.022239	62.840425	62.796197	62.715342	62.711551	62.776244
	8	85	68.774789	63.022240	62.840426	62.796197	62.715343	62.711552	62.776246
	9	60	68.393380	62.942271	62.802110	62.770522	62.745272	62.753181	62.855122
	9	65	68.392819	62.940192	62.799859	62.768214	62.742704	62.750547	62.852263
	9	70	68.392783	62.940150	62.799815	62.768168	62.742652	62.750493	62.852200
	9	75	68.392765	62.940142	62.799809	62.768164	62.742651	62.750493	62.852204
	9	80	68.392762	62.940145	62.799813	62.768168	62.742657	62.750499	62.852211
	9	85	68.392760	62.940147	62.799815	62.768170	62.742661	62.750503	62.852216
5	7	60	53.599692	51.091929	51.111406	51.126898	51.287649	51.334822	51.548855
	7	65	53.5996	51.09152	51.110966	51.126448	51.287154	51.334315	51.548307
	7	70	53.599596	51.09151	51.110956	51.126438	51.287143	51.334304	51.548296
	7	75	53.599595	51.091509	51.110955	51.126437	51.287143	51.334304	51.548295
	7	80	53.599594	51.091509	51.110956	51.126438	51.287143	51.334305	51.548296
	7	85	53.599594	51.09151	51.110956	51.126438	51.287144	51.334305	51.548297
	8	60	53.332281	51.086752	51.145952	51.173563	51.38873	51.447382	51.698542
	8	65	53.332029	51.085751	51.144879	51.172465	51.387521	51.446144	51.697202
	8	70	53.33202	**51.085735**	51.144864	51.17245	51.387505	51.446128	51.697184
	8	75	53.332016	51.085735	51.144864	51.17245	51.387507	51.44613	51.697187
	8	80	53.332016	51.085737	51.144866	51.172452	51.38751	51.446133	51.697191
	8	85	53.332015	51.085738	51.144867	51.172454	51.387511	51.446134	51.697192
	9	60	53.113941	51.128589	51.221567	51.259361	51.519826	51.587903	51.869397
	9	65	53.113399	51.126636	51.219479	51.257226	51.517481	51.585502	51.866807
	9	70	53.113382	51.126623	51.219466	51.257212	51.517466	51.585486	51.866787
	9	75	53.113376	51.126628	51.219473	51.25722	51.517477	51.585498	51.866802
	9	80	53.113374	51.126633	51.219479	51.257226	51.517485	51.585506	51.866811
	9	85	53.113374	51.126636	51.219482	51.25723	51.517489	51.58551	51.866816
7	6	60	52.477184	49.981219	49.978552	49.987232	50.114782	50.155128	50.345622
	6	65	52.477154	49.981085	49.978409	49.987086	50.114625	50.154968	50.345451
	6	70	52.477152	49.981080	49.978404	49.987082	50.114620	50.154963	50.345447
	6	75	52.477152	49.981079	49.978404	49.987081	50.114620	50.154963	50.345446
	6	80	52.477152	49.981080	49.978404	49.987081	50.114620	50.154963	50.345446
	6	85	52.477152	49.981080	49.978404	49.987081	50.114620	50.154963	50.345446

	7	60	52.157695	49.912355	49.955886	49.978858	50.171456	50.225743	50.461679
	7	65	52.157579	49.911912	49.955416	49.978378	50.170934	50.225209	50.461105
	7	70	52.157573	49.911900	49.955404	49.978366	50.170922	50.225197	50.461093
	7	75	52.157572	**49.911899**	49.955403	49.978366	50.170922	50.225197	50.461093
	7	80	52.157572	49.911899	49.955404	49.978366	50.170922	50.225197	50.461094
	7	85	52.157572	49.911899	49.955404	49.978366	50.170923	50.225198	50.461094
	8	60	51.896937	49.915920	49.998791	50.033748	50.280144	50.345759	50.618335
	8	65	51.896625	49.914856	49.997660	50.032593	50.278884	50.344470	50.616946
	8	70	51.896612	49.914836	49.997641	50.032575	50.278865	50.344450	50.616925
	8	75	51.896609	49.914836	49.997642	50.032576	50.278867	50.344453	50.616928
	8	80	51.896608	49.914838	49.997644	50.032578	50.278870	50.344456	50.616932
	8	85	51.896608	49.914839	49.997645	50.032579	50.278871	50.344457	50.616934
9	6	60	52.363551	49.887346	49.887206	49.896706	50.027893	50.069093	50.262328
	6	65	52.363518	49.887211	49.887062	49.89656	50.027735	50.068932	50.262156
	6	70	52.363516	49.887206	49.887057	49.896555	50.02773	50.068927	50.262151
	6	75	52.363516	49.887206	49.887057	49.896554	50.02773	50.068926	50.26215
	6	80	52.363516	49.887206	49.887057	49.896554	50.02773	50.068926	50.26215
	6	85	52.363516	49.887206	49.887057	49.896555	50.02773	50.068926	50.26215
	7	60	52.041548	49.815761	49.861755	49.885524	50.08166	50.136774	50.375372
	7	65	52.041426	49.815316	49.861283	49.885042	50.081138	50.136239	50.374798
	7	70	52.04142	49.815304	49.861271	49.88503	50.081125	50.136227	50.374785
	7	75	52.041419	**49.815302**	49.86127	49.885028	50.081124	50.136226	50.374784
	7	80	52.041419	49.815302	49.86127	49.885029	50.081125	50.136226	50.374785
	7	85	52.041419	49.815303	49.86127	49.885029	50.081125	50.136227	50.374785
	8	60	51.778548	49.816911	49.902194	49.937929	50.187797	50.25422	50.529408
	8	65	51.778227	49.815844	49.901061	49.936773	50.186536	50.252931	50.52802
	8	70	51.778213	49.815823	49.901041	49.936752	50.186515	50.252909	50.527996
	8	75	51.77821	49.815822	49.90104	49.936752	50.186516	50.25291	50.527999
	8	80	51.778209	49.815823	49.901042	49.936754	50.186518	50.252912	50.528002
	8	85	51.778208	49.815824	49.901043	49.936755	50.186519	50.252914	50.528003

Table 13: Perplexity improvements of the Weighted Exponential Model

***n*-gram**	***d***	**Cache**	**Sqrt(T_i)**	**LnT_i**	**Sqrt(LnT_i)**	**Ln(1+LnT_i)**	**1/Ln(1+LnT_i)**	**Sqrt(1/LnT_i)**	**1/LnT_i**
3	7	60	8.125527%	16.153987%	16.460308%	16.538964%	16.734871%	16.758771%	16.733260%
	7	65	8.125654%	16.154542%	16.460918%	16.539591%	16.735581%	16.759502%	16.734061%
	7	70	8.125662%	16.154561%	16.460938%	16.539611%	16.735602%	16.759524%	16.734084%
	7	75	8.125666%	16.154565%	16.460942%	16.539615%	16.735606%	16.759528%	16.734087%
	7	80	8.125666%	16.154565%	16.460941%	16.539615%	16.735605%	16.759526%	16.734087%
	7	85	8.125668%	16.154565%	16.460941%	16.539614%	16.735605%	16.759526%	16.734086%
	8	60	8.737941%	16.370316%	16.611450%	16.670098%	16.777199%	16.782180%	16.696166%
	8	65	8.738287%	16.371708%	16.612968%	16.671657%	16.778948%	16.783977%	16.698129%
	8	70	8.738310%	16.371744%	16.613007%	16.671697%	16.778992%	16.784022%	16.698179%
	8	75	8.738320%	16.371752%	16.613014%	16.671704%	16.778996%	**16.784026%**	16.698182%
	8	80	8.738323%	16.371752%	16.613012%	16.671701%	16.778993%	16.784023%	16.698178%
	8	85	8.738324%	16.371750%	16.613011%	16.671701%	16.778992%	16.784022%	16.698175%

	9	60	9.244440%	16.477866%	16.663855%	16.705771%	16.739277%	16.728782%	16.593510%
	9	65	9.245185%	16.480625%	16.666842%	16.708834%	16.742684%	16.732277%	16.597304%
	9	70	9.245232%	16.480681%	16.666900%	16.708895%	16.742753%	16.732349%	16.597387%
	9	75	9.245256%	16.480691%	16.666908%	16.708900%	16.742755%	16.732349%	16.597382%
	9	80	9.245260%	16.480687%	16.666903%	16.708895%	16.742747%	16.732341%	16.597373%
	9	85	9.245263%	16.480685%	16.666900%	16.708892%	16.742742%	16.732335%	16.597366%
5	7	60	28.875133%	32.202845%	32.176999%	32.156442%	31.943131%	31.880534%	31.596520%
	7	65	28.875255%	32.203387%	32.177583%	32.157039%	31.943788%	31.881207%	31.597247%
	7	70	28.875261%	32.203401%	32.177596%	32.157052%	31.943803%	31.881222%	31.597262%
	7	75	28.875262%	32.203402%	32.177598%	32.157054%	31.943803%	31.881222%	31.597263%
	7	80	28.875263%	32.203402%	32.177596%	32.157052%	31.943803%	31.881220%	31.597262%
	7	85	28.875263%	32.203401%	32.177596%	32.157052%	31.943801%	31.881220%	31.597261%
	8	60	29.229978%	32.209714%	32.131158%	32.094519%	31.809000%	31.731171%	31.397891%
	8	65	29.230312%	32.211043%	32.132582%	32.095976%	31.810605%	31.732814%	31.399669%
	8	70	29.230324%	**32.211064%**	32.132602%	32.095996%	31.810626%	31.732835%	31.399693%
	8	75	29.230330%	32.211064%	32.132602%	32.095996%	31.810623%	31.732833%	31.399689%
	8	80	29.230330%	32.211061%	32.132599%	32.095993%	31.810619%	31.732829%	31.399684%
	8	85	29.230331%	32.211060%	32.132598%	32.095991%	31.810618%	31.732827%	31.399682%
	9	60	29.519707%	32.154198%	32.030820%	31.980668%	31.635041%	31.544705%	31.171172%
	9	65	29.520426%	32.156790%	32.033590%	31.983501%	31.638152%	31.547891%	31.174609%
	9	70	29.520449%	32.156807%	32.033608%	31.983520%	31.638172%	31.547912%	31.174636%
	9	75	29.520457%	32.156800%	32.033598%	31.983509%	31.638158%	31.547896%	31.174616%
	9	80	29.520460%	32.156794%	32.033590%	31.983501%	31.638147%	31.547886%	31.174604%
	9	85	29.520460%	32.156790%	32.033586%	31.983496%	31.638142%	31.547880%	31.174597%
7	6	60	30.364661%	33.676717%	33.680256%	33.668738%	33.499484%	33.445946%	33.193167%
	6	65	30.364701%	33.676895%	33.680446%	33.668932%	33.499692%	33.446158%	33.193394%
	6	70	30.364703%	33.676901%	33.680452%	33.668937%	33.499699%	33.446165%	33.193399%
	6	75	30.364703%	33.676903%	33.680452%	33.668938%	33.499699%	33.446165%	33.193401%
	6	80	30.364703%	33.676901%	33.680452%	33.668938%	33.499699%	33.446165%	33.193401%
	6	85	30.364703%	33.676901%	33.680452%	33.668938%	33.499699%	33.446165%	33.193401%
	7	60	30.788611%	33.768097%	33.710333%	33.679850%	33.424279%	33.352242%	33.039164%
	7	65	30.788765%	33.768685%	33.710957%	33.680487%	33.424972%	33.352951%	33.039925%
	7	70	30.788773%	33.768701%	33.710972%	33.680503%	33.424988%	33.352967%	33.039941%
	7	75	30.788775%	**33.768702%**	33.710974%	33.680503%	33.424988%	33.352967%	33.039941%
	7	80	30.788775%	33.768702%	33.710972%	33.680503%	33.424988%	33.352967%	33.039940%
	7	85	30.788775%	33.768702%	33.710972%	33.680503%	33.424986%	33.352966%	33.039940%
	8	60	31.134628%	33.763366%	33.653399%	33.607013%	33.280054%	33.192985%	32.831287%
	8	65	31.135042%	33.764778%	33.654900%	33.608545%	33.281726%	33.194696%	32.833130%
	8	70	31.135059%	33.764805%	33.654925%	33.608569%	33.281751%	33.194722%	32.833158%
	8	75	31.135063%	33.764805%	33.654924%	33.608568%	33.281749%	33.194718%	32.833154%
	8	80	31.135064%	33.764802%	33.654921%	33.608565%	33.281745%	33.194714%	32.833149%
	8	85	31.135064%	33.764801%	33.654920%	33.608564%	33.281743%	33.194713%	32.833146%
9	6	60	30.515448%	33.801283%	33.801469%	33.788863%	33.614782%	33.560111%	33.303695%
	6	65	30.515492%	33.801462%	33.801660%	33.789056%	33.614992%	33.560325%	33.303924%
	6	70	30.515494%	33.801469%	33.801666%	33.789063%	33.614998%	33.560332%	33.303930%
	6	75	30.515494%	33.801469%	33.801666%	33.789064%	33.614998%	33.560333%	33.303932%

6	80	30.515494%	33.801469%	33.801666%	33.789064%	33.614998%	33.560333%	33.303932%
6	85	30.515494%	33.801469%	33.801666%	33.789063%	33.614998%	33.560333%	33.303932%
7	60	30.942734%	33.896274%	33.835241%	33.803701%	33.543435%	33.470301%	33.153690%
7	65	30.942896%	33.896864%	33.835868%	33.804340%	33.544128%	33.471011%	33.154452%
7	70	30.942904%	33.896880%	33.835884%	33.804356%	33.544145%	33.471027%	33.154469%
7	75	30.942905%	**33.896883%**	33.835885%	33.804359%	33.544147%	33.471028%	33.154470%
7	80	30.942905%	33.896883%	33.835885%	33.804358%	33.544145%	33.471028%	33.154469%
7	85	30.942905%	33.896881%	33.835885%	33.804358%	33.544145%	33.471027%	33.154469%
8	60	31.291726%	33.894748%	33.781580%	33.734161%	33.402595%	33.314454%	32.949290%
8	65	31.292152%	33.896164%	33.783084%	33.735695%	33.404269%	33.316165%	32.951132%
8	70	31.292170%	33.896191%	33.783110%	33.735723%	33.404296%	33.316194%	32.951163%
8	75	31.292174%	33.896193%	33.783112%	33.735723%	33.404295%	33.316193%	32.951159%
8	80	31.292176%	33.896191%	33.783109%	33.735720%	33.404292%	33.316190%	32.951155%
8	85	31.292177%	33.896190%	33.783108%	33.735719%	33.404291%	33.316187%	32.951154%

Our overall results combined to smaller corpus-sizes (Sicilia-Garcia et al., 2001, 2002) in Table 14 (their WSJ88 and WSJ89 results were not for 7-grams.)

Table 14: Resume for the Weighted Exponential Model

		Sentence/WSJ			**Sentence/WSJ88**			**Sentence/WSJ89**		
		Perplexity	**Improvement**	***(d, Cache, Function)***	**Perplexity**	**Improve-Ment**	***(d, Cache, Function)***	**Perplexity**	**Improvement**	***(d, Cache, Function)***
***n*-gram**	3	62.71	16.78%	*(8, 75, Sqrt(1/LnT$_i$))*	74.76	13.60%	*(8, 45, LnT$_i$)*	96.36	10.07%	*(7, 55, LnT$_i$)*
	5	51.09	32.21%	*(8, 70, LnT$_i$)*	64.05	25.98%	*(7, 45, LnT$_i$)*	87.49	18.37%	*(6, 55, LnT$_i$)*
	7	49.91	33.77%	*(7, 75, LnT$_i$)*	-	-	-	-	-	-
	9	49.82	33.90%	*(7, 75, LnT$_i$)*	63.27	26.87%	*(7, 45, LnT$_i$)*	87.34	18.51%	*(6, 50, LnT$_i$)*

An improvement of 34% has been achieved for the weighted exponential model. The optimum weight is LnT_i. The exponential distance decay is 7 and the cache is up to 75 words.

6.3.3. Linear Interpolation Exponential Model with weights

Our total results combined to smaller corpus-sizes of previous authors (Sicilia-Garcia et al., 2001, 2002) in Table 15 (Their WSJ88 and WSJ89 results were not for 7-grams.)

Table 15: Improvement in perplexity for the linear interpolation exponential model with weights

		Sentence/WSJ			**Sentence/WSJ88**			**Sentence/WSJ89**		
		Perplexity	**Improvement**	***(λ, d, Cache, Function)***	**Perplexity**	**Improve-Ment**	***(λ, d, Cache, Function)***	**Perplexity**	**Improvement**	***(λ, d, Cache, Function)***
***n*-gram**	3	65.63	12.91%	*(0.4, 8, 75, Sqrt(1/LnT$_i$))*	77.51	10.42%	*(0.5, 9, 60, Ln(1+LnT$_i$))*	98.54	8.06%	*(0.6, 13, 60, LnT$_i$)*
	5	52.51	30.32%	*(0.6, 11, 70, Sqrt(1/LnT$_i$))*	65.37	24.46%	*(0.7, 15, 60, Ln(1+LnT$_i$))*	88.17	17.74%	*(0.7, 20, 60, LnT$_i$)*
	7	51.16	32.11%	*(0.6, 13, 65, Ln(1+LnT$_i$))*	-	-	-	-	-	-
	9	51.03	32.28%	*(0.6, 13, 65, Ln(1+LnT$_i$))*	64.36	25.63%	*(0.7, 18, 65, Ln(1+LnT$_i$))*	87.85	18.04%	*(0.7, 23, 65, LnT$_i$)*

Our best perplexity improvement for this kind of individual word models is over 32%, slightly lower than the best result 34% of the weighted exponential model. These results are disappointing since this model is a combination of all the models described previously. Also the time of execution is longer than any of the methods described above.

6.4. Best results of the probability models

The best performance of the probability models is shown by Table 16.

Table 16: Improvement in perplexity for different combinations of word models, all the results correspond to the combined WSJ corpus in sentence contexts

Models	tri-gram	5-gram	7-gram	9-gram	Comment. Best Values
Global Model	0.00%	24.57%	26.63%	26.77%	
Linear Interpolation Model	11.16%	29.84%	31.78%	31.98%	λ=0.7, WM=23
Exponential Decay Model	16.75%	32.09%	33.61%	33.72%	Decay=6, Cache=70
Weighted Probability Model	13.90%	30.55%	32.35%	32.52%	WM=16, $Sqrt(T_i)$
Weighted Exponential Model	16.78%	32.21%	33.77%	33.90%	Decay=7, Cache=75, LnT_i
Linear Interpolation Exponential with weights	12.91%	30.32%	32.11%	32.28%	λ=0.6, Decay=13, Cache=65, $Ln(1+LnT_i)$

The best model is the weighted exponential model with 34% improvement. For these models, the number of individual word probability models in the cache to reach the maximum performance is from 16 to 23. So the individual word-domain language model reduces the size of the cache needed from 500 words as in other models (Clarkson and Robinson, 1997; Donnelly, Smith, Sicilia-Garcia and Ming, 1999) to less than 30 words, which is important for spoken language and closer to the ability of humans.

7. Conclusions

We have introduced the concept of individual word probability models to improve language model performance. Individual word language models permit an accurate capture of the domains in which significant words occur and hence improve the language model performance. The results indicate that individual word models offer a promising and simple means of introducing domain information into an *n*-gram language model. The improvement in perplexity so far (34%) in Table 16 is better to that obtained in much more computationally intensive methods based on clustering (Iyer and Ostendorf, 1999; Clarkson et al., 1997).

For human, we feel that these *a priori* results are more probably accurate than the weighted average model. Humans probably hear the sounds of several words spoken before using a form of human language model to make a sensible sentence from the sounds, particularly when there are corruptions. So the idea of using all the words in a sentence in order to define the domain might be more appropriate than the *a priori* method. Sicilia-Garcia, Ming and Smith (2005) first tried this idea as the *a posteriori* method for the weighted probability model and predicted a 68%-69% perplexity improvement but they could not give a full explanation, hence they were not sure. In our preliminary experiments, we are now re-testing the work more in details. It shows that their method needs a necessary adjustment that will bring more practical, more reasonable and more explicable results; these will be our future work. Also word error rate measurements are needed.

Acknowledgements

The authors would like to thank Dr Philip Hanna for his support and for his valuable comments.

References

[1] Clarkson, P. R. and Robinson, A. J. "*Language Model Adaptation Using Mixtures and an Exponentially Decaying Cache*". IEEE ICASSP'97, Vol. 2, pp. 799-802. Munich, Germany. 1997.

[2] Donnelly, P. "*A Domain Based Approach to Natural Language Modelling*". PhD Thesis. Queen's University Belfast, Northern Ireland. September 1998.

[3] Donnelly, P. G., Smith, F. J, Sicilia-Garcia, E. I. and Ming, J. "*Language Modelling With Hierarchical Domains"*. Proceeding of Eurospeech'99, Vol. 4, pp. 1575-1578. Budapest, Hungary. 1999.

[4] Good, I. J. *"The Population Frequencies of Species and the Estimation of Population Parameters"*. Biometrika, Vol. 40, pp. 237-254. 1953.

[5] Iyer, R. M. and Ostendorf, M. "*Modeling Long Distance Dependence in Language: Topic Mixture Versus Dynamic Cache Models*". IEEE Transactions on Speech and Audio Processing, Vol. 17, No 1, pp. 30-39. 1999.

[6] Jelinek, F., Mercer, R. L. and Bahl, L. R. "*A Maximum Likelihood Approach to Continuous Speech Recognition*". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 5, pp. 179-190. 1983.

[7] Katz, S. M. "*Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recogniser*". IEEE Transactions On Acoustic Speech and Signal Processing, Vol. 35 (3), pp. 400-401. 1987.

[8] Kuhn, R. and De Mori, R. "*A Cache-Based Natural Language Model for Speech Recognition"*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 12 (6), pp. 570-583. 1990.

[9] Lau, R., Rosenfeld, R. and Roukos, S. "*Trigger-based Language models: A Maximum entropy approach*". IEEE ICASSP'93, Vol. 2, pp. 45-48. Minneapolis, MN, USA. 1993.

[10] O'Boyle, P., Owens, M. and Smith, F. J. *"Average n-gram Model of Natural Language"*. Computer Speech and Language, Vol. 8, pp. 337-349. 1994.

[11] Paul, D. B. and Baker, J. M. "*The Design for the Wall Street Journal-based CSR corpus*", Proceeding of ICLSP'92, pp. 899-902. November 1992.

[12] Seymore, K., Chen, S. and Rosenfeld, R. "*Nonlinear Interpolation of Topic Models for Language Model Adaptation*". ICSLP'98, Vol. 6, pp. 2503-2506. Sydney, Australia. December 1998.

[13] Sicilia-Garcia, E. I. *"A Study in Dynamic Language Modelling"*. PhD Thesis. Queen's University Belfast, Northern Ireland. 2002.

[14] Sicilia-Garcia, E. I., Ming, J. and Smith, F. J. *"A posteriori multiple word-domain language model"*. Interspeech-Eurospeech'05, pp. 1285-1288. Lisbon, Portugal. September 2005.

[15] Sicilia-Garcia, E. I., Ming, J. and Smith, F. J. "*Individual Word Language Models and the Frequency Approach*". ICSLP'02, pp. 897-900. Denver, Colorado. September 2002.

[16] Sicilia-Garcia, E. I., Ming, J. and Smith, F. J. "*Triggering Individual Word Domains in n-gram Language Models*". Eurospeech'01, Vol. 1, pp 701-704. Aalborg, Denmark. September 2001.