Одним из способов повышения эффективности работы систем информационного поиска является предоставление поисковым системам способа обнаружения различных форм одного и того же слова. Для реализации процесса обнаружения различных форм можно воспользоваться стеммерами. Стемминг также используется в информационном поиске для уменьшения размера индексных файлов. Таким образом сначала для исходного текста выделяются границы. Затем для обнаружения различных форм одного и того же слова необходимо выполнить процесс нахождения основы слова для всех исходных слов текста – стемминг. Стемминг выполняет морфологический разбор слова, находит общую для всех его грамматических форм основу, отсекая суффиксы и окончания.

Существует несколько критериев оценки стеммеров: корректность, эффективность поиска и производительность сжатия.

При реализации стемминга нужно найти баланс между следующими двумя проблемами: чрезмерный стемминг, что приводит к объединению несвязанных терминов и соответственно это понижает точность поиска, так как извлекаются нерелевантные документы; основа слова выделяется слишком слабо, в связи с чем будет понижаться полнота поиска.

В зависимости от необходимой точности/полноты поиска, а также скорости работы можно выбрать один из следующих стеммеров.

Для английского языка на данный момент одним из самых распространённых стеммеров является стеммер Портера в силу его быстрой скорости работы, отсутствия необходимости в предварительной обработке корпуса документов и использования каких-либо баз основ. В основе данного стеммера лежит алгоритм усечения окончаний, использующий для своей работы небольшой набор правил, например, если слово оканчивается на “ет”, то удалить “eт” и так далее. Алгоритмы усечения окончаний достаточно эффективны на практике, но в то же время обладают некоторыми недостатками. Алгоритмы усечения окончаний неэффективны в случае изменения корня слова, например, изменения или выпадения гласной. Данные алгоритмы эффективны для тех частей речи, которые имеют хорошо известные окончания и суффиксы. Стеммер Портера основывается на том, что количество словообразующих суффиксов в языках ограниченно. Благодаря этому алгоритм может выполняться с помощью установленных вручную определённых правил. Алгоритм выделения основы слова стеммера Портера для английского языка состоит из пяти шагов. На каждом шаге у слова отсекается словообразующий суффикс, затем оставшаяся часть проверяется на соответствие заданным правилам. В случае, если правила удовлетворены осуществляется переход на следующий шаг алгоритма, иначе выбирается другой суффикс для отсечения. Из описания хода работы алгоритма видно, что у стеммера Портера существует недостаток: он может обрезать слово больше необходимого, что в свою очередь затруднит получение правильной основы слова и соответственно уменьшит точность извлечение релевантной информации. Ещё одним недостатком стеммера Портера является отсутствие возможности работать при изменении корня слова, например, в случае выпадающих беглых гласных.

Стеммер, использующий таблицы поиска флективных норм. Трудностью при реализации данного стеммера является необходимость перечислять все флективные формы в таблице, если какая-то из форм будет отсутствовать, то она обрабатываться не будет. В связи с этим получается, что таблица поиска может иметь большой размер. В качестве плюсов можно выделить простоту подхода, скорость работы и простоту обработки исключений. Таблицы поиска, которые используются в стеммерах, обычно генерируются в полуавтоматическом режиме. Чтобы избежать проблемы, когда разные слова относятся к одной лемме (ошибка лемматизации), при реализации алгоритма поиска можно использовать предварительную частеречную разметку.

Можно улучшить проход к выделению основы слова посредством определения части речи слова и затем в зависимости от результата применения соответствующих для каждой части речи правил нормализации.

Основной недостаток классических стеммеров – они не различают слова, имеющие схожий синтаксис, но абсолютно разные значения, например, в английском языке “news” и “new” для данных стеммеров будут различными формами одного и того же слова. С целью разрешения данных проблем было реализованы стеммеры на основе корпусов текстов. Ключевой идеей их является создании классов эквивалентности для слов классических стеммеров, которые после разделят некоторые слова, объединенные на основе их встречаемости в корпусе. Такие алгоритмы работают с базой данных основ, которые не обязательно соответствуют обычным словам и зачастую представляют собой. Для определения основы слова алгоритм сопоставляет его с основами из базы данных, используя различные ограничения, такие как длина искомой основы в слове относительно длины самого слова и т.п.